



Welcome to [E-XFL.COM](https://www.e-xfl.com)

What is "[Embedded - Microcontrollers](#)"?

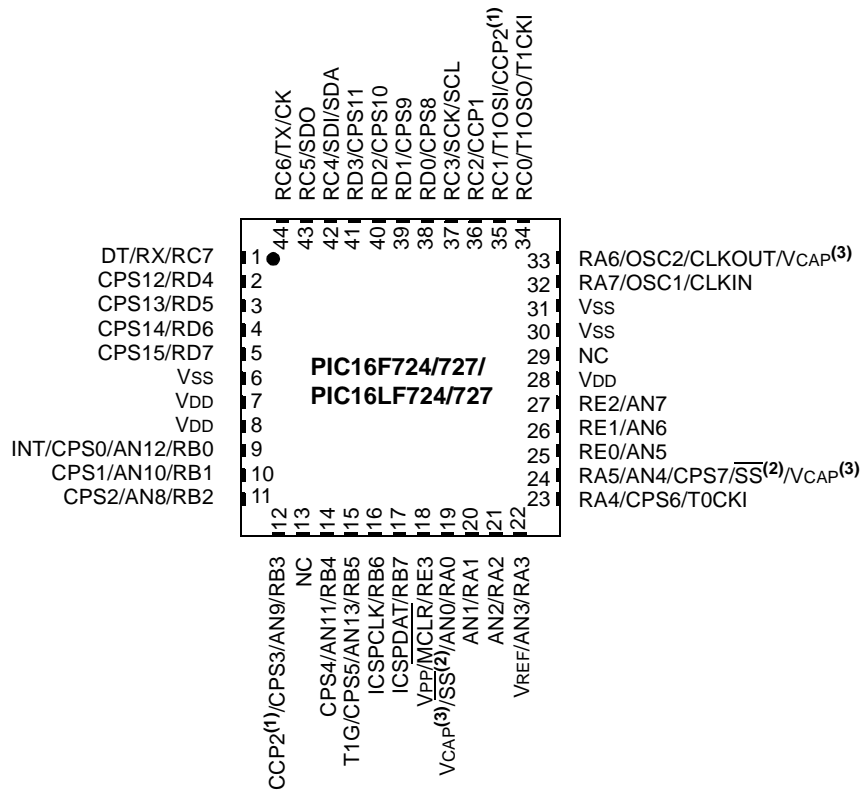
"[Embedded - Microcontrollers](#)" refer to small, integrated circuits designed to perform specific tasks within larger systems. These microcontrollers are essentially compact computers on a single chip, containing a processor core, memory, and programmable input/output peripherals. They are called "embedded" 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 | 20MHz |
| Connectivity | I ² C, SPI, UART/USART |
| Peripherals | Brown-out Detect/Reset, POR, PWM, WDT |
| Number of I/O | 25 |
| Program Memory Size | 14KB (8K x 14) |
| Program Memory Type | FLASH |
| EEPROM Size | - |
| RAM Size | 368 x 8 |
| Voltage - Supply (Vcc/Vdd) | 1.8V ~ 5.5V |
| Data Converters | A/D 11x8b |
| Oscillator Type | Internal |
| Operating Temperature | -40°C ~ 125°C (TA) |
| Mounting Type | Surface Mount |
| Package / Case | 28-UQFN Exposed Pad |
| Supplier Device Package | 28-UQFN (4x4) |
| Purchase URL | https://www.e-xfl.com/product-detail/microchip-technology/pic16f726-e-mv |

Pin Diagrams – 44-PIN QFN (PIC16F724/727/PIC16LF724/727)



- Note** 1: CCP2 pin location may be selected as RB3 or RC1.
2: SS pin location may be selected as RA5 or RA0.
3: PIC16F724/727 devices only.

PIC16(L)F722/3/4/6/7

TABLE 2-1: PIC16(L)F722/3/4/6/7 SPECIAL FUNCTION REGISTER SUMMARY

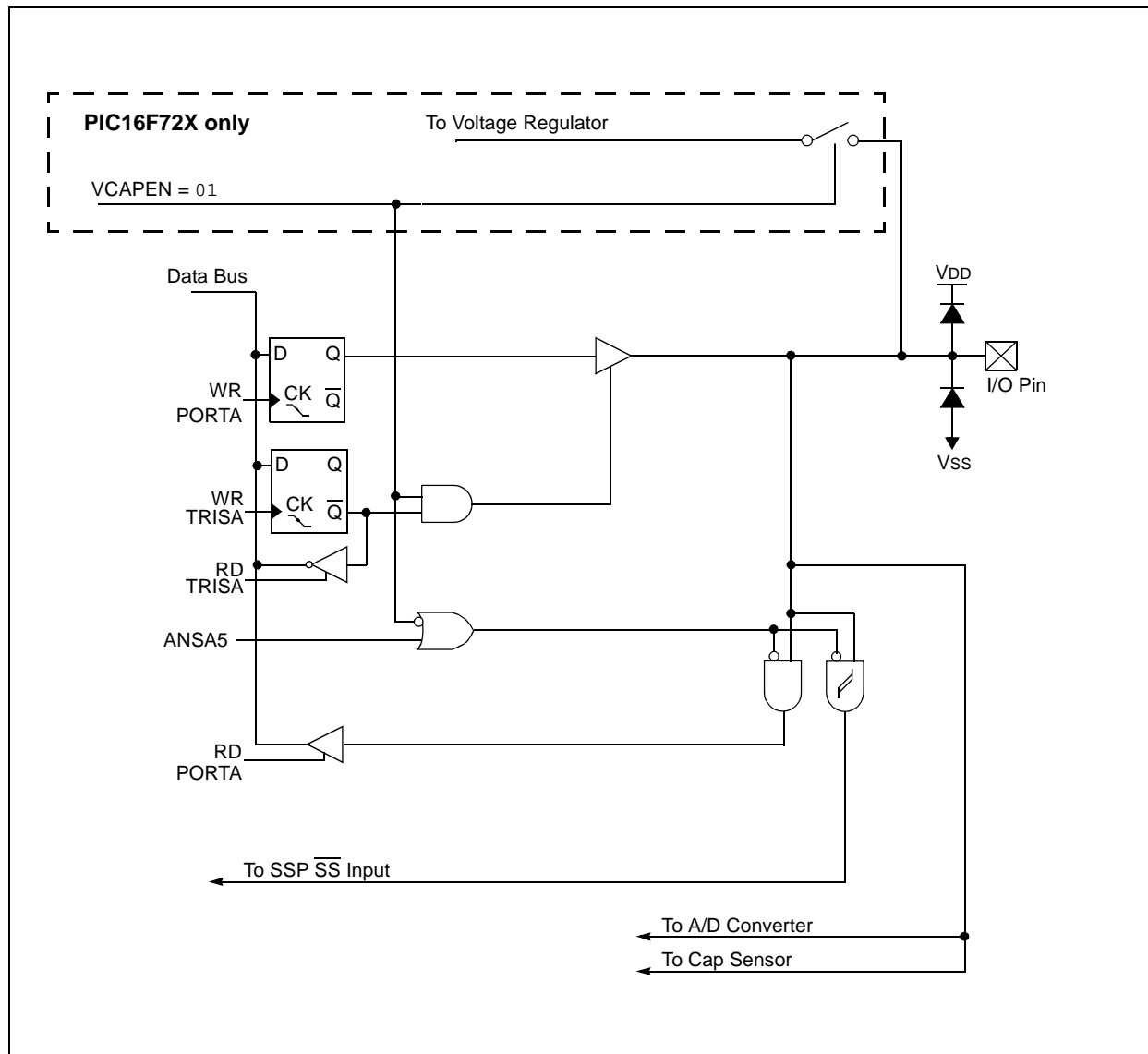
| Address | Name | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | Value on: POR, BOR | Page |
|-----------------------|---------|--|---------|---------|--|-----------------|--------------------------------|----------------------|--------------------|--------------------|--------|
| Bank 0 | | | | | | | | | | | |
| 00h ⁽²⁾ | INDF | Addressing this location uses contents of FSR to address data memory (not a physical register) | | | | | | | | xxxx xxxx | 29,37 |
| 01h | TMR0 | Timer0 Module Register | | | | | | | | xxxx xxxx | 105,37 |
| 02h ⁽²⁾ | PCL | Program Counter (PC) Least Significant Byte | | | | | | | | 0000 0000 | 28,37 |
| 03h ⁽²⁾ | STATUS | IRP | RP1 | RP0 | \overline{TO} | \overline{PD} | Z | DC | C | 0001 1xxx | 25,37 |
| 04h ⁽²⁾ | FSR | Indirect Data Memory Address Pointer | | | | | | | | xxxx xxxx | 29,37 |
| 05h | PORTA | RA7 | RA6 | RA5 | RA4 | RA3 | RA2 | RA1 | RA0 | xxxx xxxx | 51,37 |
| 06h | PORTB | RB7 | RB6 | RB5 | RB4 | RB3 | RB2 | RB1 | RB0 | xxxx xxxx | 60,37 |
| 07h | PORTC | RC7 | RC6 | RC5 | RC4 | RC3 | RC2 | RC1 | RC0 | xxxx xxxx | 70,37 |
| 08h ⁽³⁾ | PORTD | RD7 | RD6 | RD5 | RD4 | RD3 | RD2 | RD1 | RD0 | xxxx xxxx | 77,37 |
| 09h | PORTE | — | — | — | — | RE3 | RE2 ⁽³⁾ | RE1 ⁽³⁾ | RE0 ⁽³⁾ | ---- xxxx | 81,37 |
| 0Ah ^(1, 2) | PCLATH | — | — | — | Write Buffer for the upper 5 bits of the Program Counter | | | | | ---0 0000 | 28,37 |
| 0Bh ⁽²⁾ | INTCON | GIE | PEIE | TOIE | INTE | RBIE | T0IF | INTF | RBIF | 0000 000x | 44,37 |
| 0Ch | PIR1 | TMR1GIF | ADIF | RCIF | TXIF | SSPIF | CCP1IF | TMR2IF | TMR1IF | 0000 0000 | 47,37 |
| 0Dh | PIR2 | — | — | — | — | — | — | — | CCP2IF | ---- --0 | 48,37 |
| 0Eh | TMR1L | Holding Register for the Least Significant Byte of the 16-bit TMR1 Register | | | | | | | | xxxx xxxx | 113,37 |
| 0Fh | TMR1H | Holding Register for the Most Significant Byte of the 16-bit TMR1 Register | | | | | | | | xxxx xxxx | 113,37 |
| 10h | T1CON | TMR1CS1 | TMR1CS0 | T1CKPS1 | T1CKPS0 | T1OSCEN | $\overline{T1SYN\overline{C}}$ | — | TMR1ON | 0000 00-0 | 117,37 |
| 11h | TMR2 | Timer2 Module Register | | | | | | | | 0000 0000 | 120,37 |
| 12h | T2CON | — | TOUTPS3 | TOUTPS2 | TOUTPS1 | TOUTPS0 | TMR2ON | T2CKPS1 | T2CKPS0 | -000 0000 | 121,37 |
| 13h | SSPBUF | Synchronous Serial Port Receive Buffer/Transmit Register | | | | | | | | xxxx xxxx | 161,37 |
| 14h | SSPCON | WCOL | SSPOV | SSPEN | CKP | SSPM3 | SSPM2 | SSPM1 | SSPM0 | 0000 0000 | 178,37 |
| 15h | CCPR1L | Capture/Compare/PWM Register (LSB) | | | | | | | | xxxx xxxx | 130,37 |
| 16h | CCPR1H | Capture/Compare/PWM Register (MSB) | | | | | | | | xxxx xxxx | 130,37 |
| 17h | CCP1CON | — | — | DC1B1 | DC1B0 | CCP1M3 | CCP1M2 | CCP1M1 | CCP1M0 | --00 0000 | 129,37 |
| 18h | RCSTA | SPEN | RX9 | SREN | CREN | ADDEN | FERR | OERR | RX9D | 0000 000x | 148,37 |
| 19h | TXREG | USART Transmit Data Register | | | | | | | | 0000 0000 | 147,37 |
| 1Ah | RCREG | USART Receive Data Register | | | | | | | | 0000 0000 | 145,37 |
| 1Bh | CCPR2L | Capture/Compare/PWM Register 2 (LSB) | | | | | | | | xxxx xxxx | 130,37 |
| 1Ch | CCPR2H | Capture/Compare/PWM Register 2 (MSB) | | | | | | | | xxxx xxxx | 130,37 |
| 1Dh | CCP2CON | — | — | DC2B1 | DC2B0 | CCP2M3 | CCP2M2 | CCP2M1 | CCP2M0 | --00 0000 | 129,37 |
| 1Eh | ADRES | A/D Result Register | | | | | | | | xxxx xxxx | 100,37 |
| 1Fh | ADCON0 | — | — | CHS3 | CHS2 | CHS1 | CHS0 | $\overline{GO/DONE}$ | ADON | --00 0000 | 99,37 |

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

- Note** 1: The upper byte of the program counter is not directly accessible. PCLATH is a holding register for the PC<12:8>, whose contents are transferred to the upper byte of the program counter.
- 2: These registers can be addressed from any bank.
- 3: These registers/bits are not implemented on PIC16F722/723/726/PIC16LF722/723/726 devices, read as '0'.
- 4: Accessible only when SSPM<3:0> = 1001.
- 5: Accessible only when SSPM<3:0> ≠ 1001.
- 6: This bit is always '1' as RE3 is input-only.

PIC16(L)F722/3/4/6/7

FIGURE 6-4: BLOCK DIAGRAM OF RA5



PIC16(L)F722/3/4/6/7

FIGURE 6-8: BLOCK DIAGRAM OF RB4, RB<2:1>

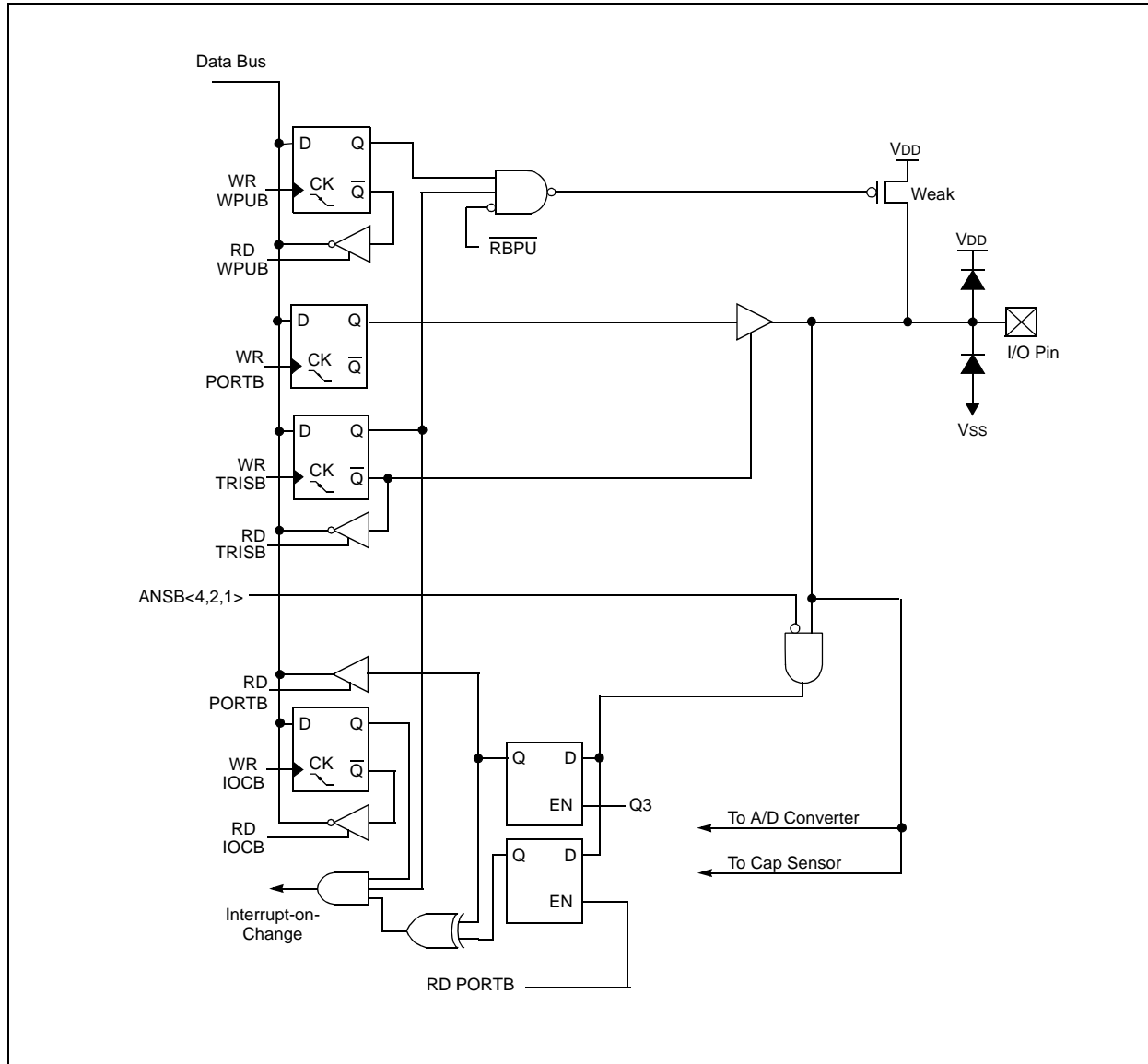




TABLE 6-2: SUMMARY OF REGISTERS ASSOCIATED WITH PORTB

| Name | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | Value on POR, BOR | Value on all other Resets |
|------------|--------|--------|--------|--------|------------|---------|---------|---------|-------------------|---------------------------|
| ADCON0 | — | — | CHS3 | CHS2 | CHS1 | CHS0 | GO/DONE | ADON | --00 0000 | --00 0000 |
| ANSELB | — | — | ANSB5 | ANSB4 | ANSB3 | ANSB2 | ANSB1 | ANSB0 | --11 1111 | --11 1111 |
| APFCON | — | — | — | — | — | — | SSSEL | CCP2SEL | ---- --00 | ---- --00 |
| CCP2CON | — | — | DC2B1 | DC2B0 | CCP2M3 | CCP2M2 | CCP2M1 | CCP2M0 | --00 0000 | --00 0000 |
| CPSCON0 | CPSON | — | — | — | CPSRNG1 | CPSRNG0 | CPSOUT | T0XCS | 0--- 0000 | 0--- 0000 |
| CPSCON1 | — | — | — | — | CPSCH3 | CPSCH2 | CPSCH1 | CPSCH0 | ---- 0000 | ---- 0000 |
| INTCON | GIE | PEIE | T0IE | INTE | RBIE | T0IF | INTF | RBIF | 0000 000x | 0000 000x |
| IOCB | IOCB7 | IOCB6 | IOCB5 | IOCB4 | IOCB3 | IOCB2 | IOCB1 | IOCB0 | 0000 0000 | 0000 0000 |
| OPTION_REG | RBPŪ | INTEDG | T0CS | T0SE | PSA | PS2 | PS1 | PS0 | 1111 1111 | 1111 1111 |
| PORTB | RB7 | RB6 | RB5 | RB4 | RB3 | RB2 | RB1 | RB0 | xxxx xxxx | xxxx xxxx |
| T1GCON | TMR1GE | T1GPOL | T1GTM | T1GSPM | T1GGO/DONE | T1GVAL | T1GSS1 | T1GSS0 | 0000 0x00 | uuuu uuuu |
| TRISB | TRISB7 | TRISB6 | TRISB5 | TRISB4 | TRISB3 | TRISB2 | TRISB1 | TRISB0 | 1111 1111 | 1111 1111 |
| WPUB | WPUB7 | WPUB6 | WPUB5 | WPUB4 | WPUB3 | WPUB2 | WPUB1 | WPUB0 | 1111 1111 | 1111 1111 |

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

PIC16(L)F722/3/4/6/7

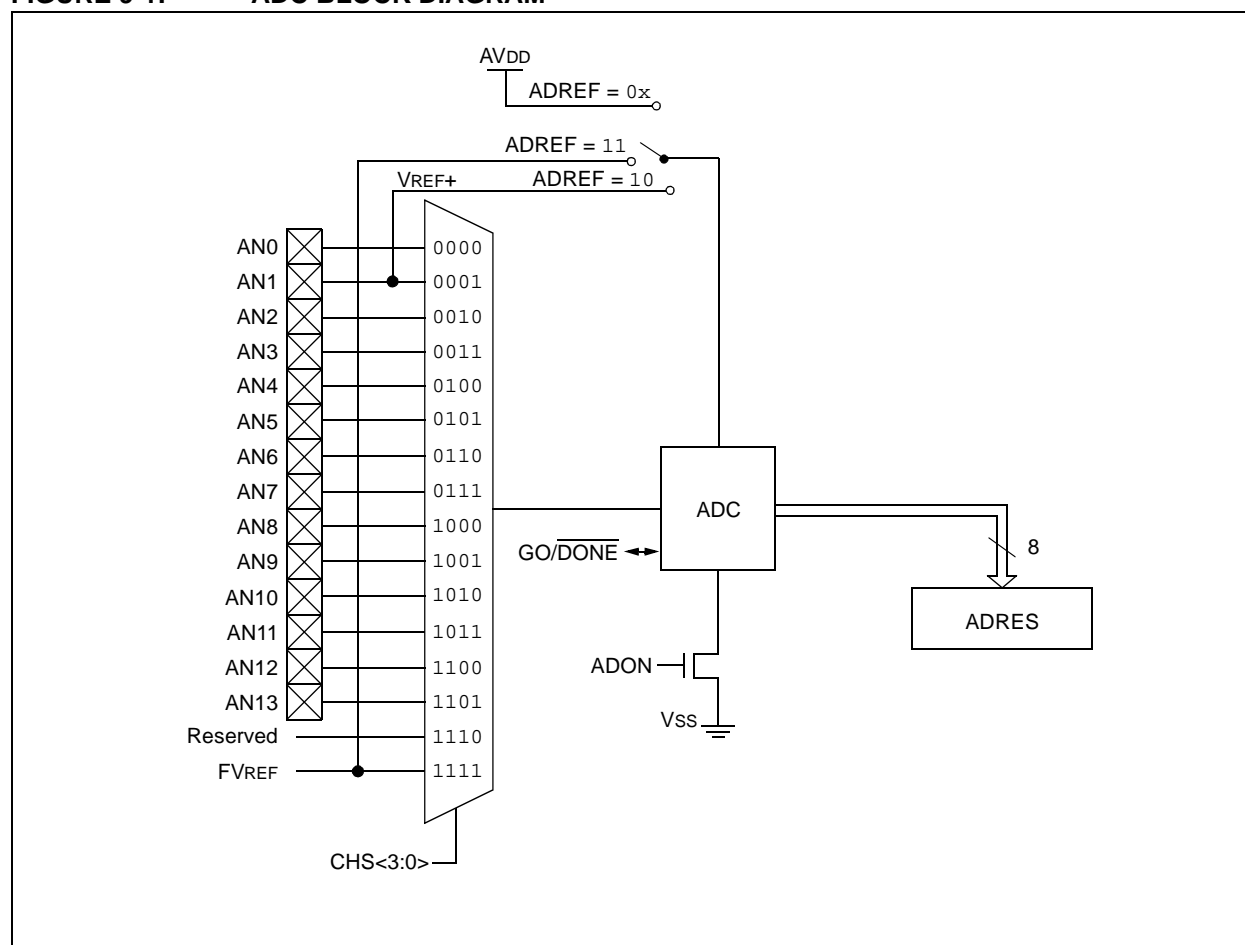
9.0 ANALOG-TO-DIGITAL CONVERTER (ADC) MODULE

The Analog-to-Digital Converter (ADC) allows conversion of an analog input signal to a 8-bit binary representation of that signal. This device uses analog inputs, which are multiplexed into a single sample and hold circuit. The output of the sample and hold is connected to the input of the converter. The converter generates a 8-bit binary result via successive approximation and stores the conversion result into the ADC result register (ADRES). Figure 9-1 shows the block diagram of the ADC.

The ADC voltage reference is software selectable to be either internally generated or externally supplied.

The ADC can generate an interrupt upon completion of a conversion. This interrupt can be used to wake-up the device from Sleep.

FIGURE 9-1: ADC BLOCK DIAGRAM



PIC16(L)F722/3/4/6/7

REGISTER 9-2: ADCON1: A/D CONTROL REGISTER 1

| | | | | | | | |
|-------|-------|-------|-------|-----|-----|--------|--------|
| U-0 | R/W-0 | R/W-0 | R/W-0 | U-0 | U-0 | R/W-0 | R/W-0 |
| — | ADCS2 | ADCS1 | ADCS0 | — | — | ADREF1 | ADREF0 |
| bit 7 | | | | | | | bit 0 |

Legend:

R = Readable bit

W = Writable bit

U = Unimplemented bit, read as '0'

-n = Value at POR

'1' = Bit is set

'0' = Bit is cleared

x = Bit is unknown

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

bit 6-4 **ADCS<2:0>:** A/D Conversion Clock Select bits

000 = FOSC/2

001 = FOSC/8

010 = FOSC/32

011 = FRC (clock supplied from a dedicated RC oscillator)

100 = FOSC/4

101 = FOSC/16

110 = FOSC/64

111 = FRC (clock supplied from a dedicated RC oscillator)

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

bit 1-0 **ADREF<1:0>:** Voltage Reference Configuration bits

0x = VREF is connected to VDD

10 = VREF is connected to external VREF (RA3/AN3)

11 = VREF is connected to internal Fixed Voltage Reference

REGISTER 9-3: ADRES: ADC RESULT REGISTER

| | | | | | | | |
|--------|--------|--------|--------|--------|--------|--------|--------|
| R/W-x | R/W-x | R/W-x | R/W-x | R/W-x | R/W-x | R/W-x | R/W-x |
| ADRES7 | ADRES6 | ADRES5 | ADRES4 | ADRES3 | ADRES2 | ADRES1 | ADRES0 |
| bit 7 | | | | | | | bit 0 |

Legend:

R = Readable bit

W = Writable bit

U = Unimplemented bit, read as '0'

-n = Value at POR

'1' = Bit is set

'0' = Bit is cleared

x = Bit is unknown

bit 7-0 **ADRES<7:0>:** ADC Result Register bits
8-bit conversion result.

PIC16(L)F722/3/4/6/7

16.0 ADDRESSABLE UNIVERSAL SYNCHRONOUS ASYNCHRONOUS RECEIVER TRANSMITTER (AUSART)

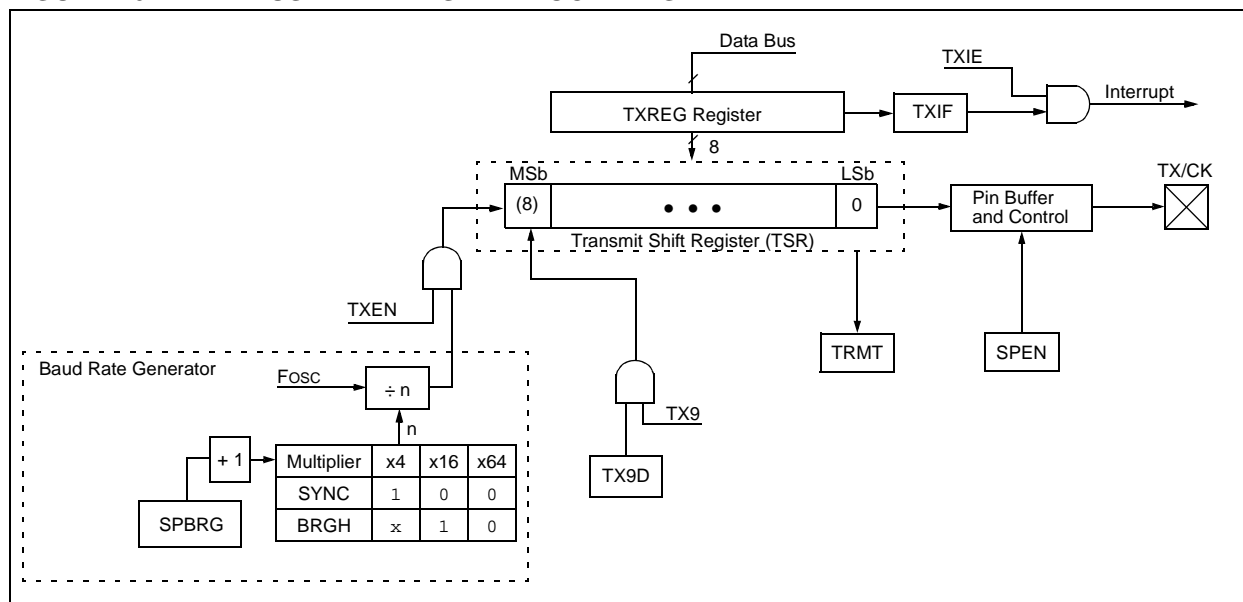
The Addressable Universal Synchronous Asynchronous Receiver Transmitter (AUSART) module is a serial I/O communications peripheral. It contains all the clock generators, shift registers and data buffers necessary to perform an input or output serial data transfer independent of device program execution. The AUSART, also known as a Serial Communications Interface (SCI), can be configured as a full-duplex asynchronous system or half-duplex synchronous system. Full-Duplex mode is useful for communications with peripheral systems, such as CRT terminals and personal computers. Half-Duplex Synchronous mode is intended for communications with peripheral devices, such as A/D or D/A integrated circuits, serial EEPROMs or other microcontrollers. These devices typically do not have internal clocks for baud rate generation and require the external clock signal provided by a master synchronous device.

The AUSART module includes the following capabilities:

- Full-duplex asynchronous transmit and receive
- Two-character input buffer
- One-character output buffer
- Programmable 8-bit or 9-bit character length
- Address detection in 9-bit mode
- Input buffer overrun error detection
- Received character framing error detection
- Half-duplex synchronous master
- Half-duplex synchronous slave
- Sleep operation

Block diagrams of the AUSART transmitter and receiver are shown in Figure 16-1 and Figure 16-2.

FIGURE 16-1: AUSART TRANSMIT BLOCK DIAGRAM



PIC16(L)F722/3/4/6/7

TABLE 16-2: REGISTERS ASSOCIATED WITH ASYNCHRONOUS RECEPTION

| Name | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | Value on POR, BOR | Value on all other Resets |
|--------|------------------------------|--------|--------|--------|--------|--------|--------|--------|-------------------|---------------------------|
| INTCON | GIE | PEIE | T0IE | INTE | RBIE | T0IF | INTF | RBIF | 0000 000x | 0000 000x |
| PIE1 | TMR1GIE | ADIE | RCIE | TXIE | SSPIE | CCP1IE | TMR2IE | TMR1IE | 0000 0000 | 0000 0000 |
| PIR1 | TMR1GIF | ADIF | RCIF | TXIF | SSPIF | CCP1IF | TMR2IF | TMR1IF | 0000 0000 | 0000 0000 |
| RCREG | AUSART Receive Data Register | | | | | | | | 0000 0000 | 0000 0000 |
| RCSTA | SPEN | RX9 | SREN | CREN | ADDEN | FERR | OERR | RX9D | 0000 000x | 0000 000x |
| SPBRG | BRG7 | BRG6 | BRG5 | BRG4 | BRG3 | BRG2 | BRG1 | BRG0 | 0000 0000 | 0000 0000 |
| TRISC | TRISC7 | TRISC6 | TRISC5 | TRISC4 | TRISC3 | TRISC2 | TRISC1 | TRISC0 | 1111 1111 | 1111 1111 |
| TXSTA | CSRC | TX9 | TXEN | SYNC | — | BRGH | TRMT | TX9D | 0000 -010 | 0000 -010 |

Legend: x = unknown, - = unimplemented read as '0'. Shaded cells are not used for Asynchronous Reception.

FIGURE 16-6: SYNCHRONOUS TRANSMISSION

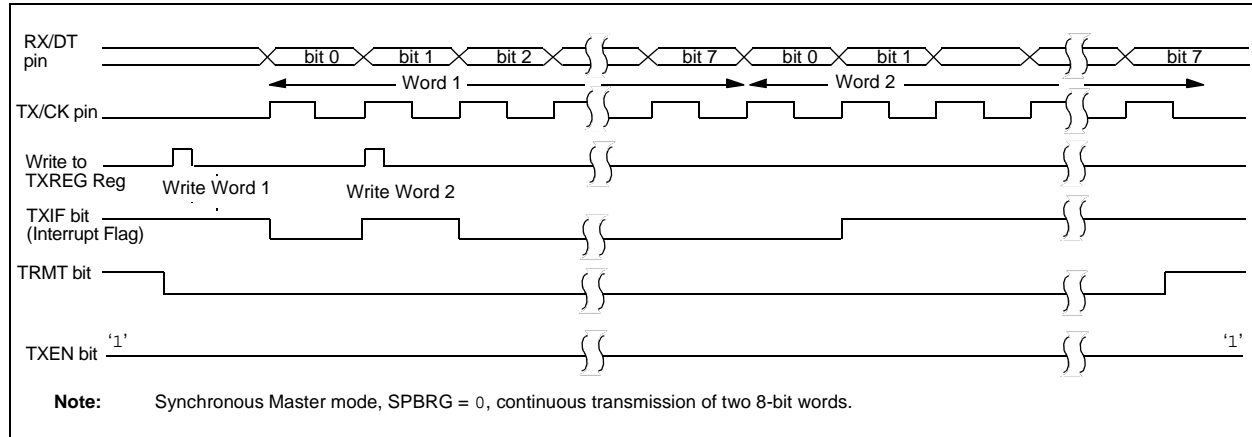


FIGURE 16-7: SYNCHRONOUS TRANSMISSION (THROUGH TXEN)

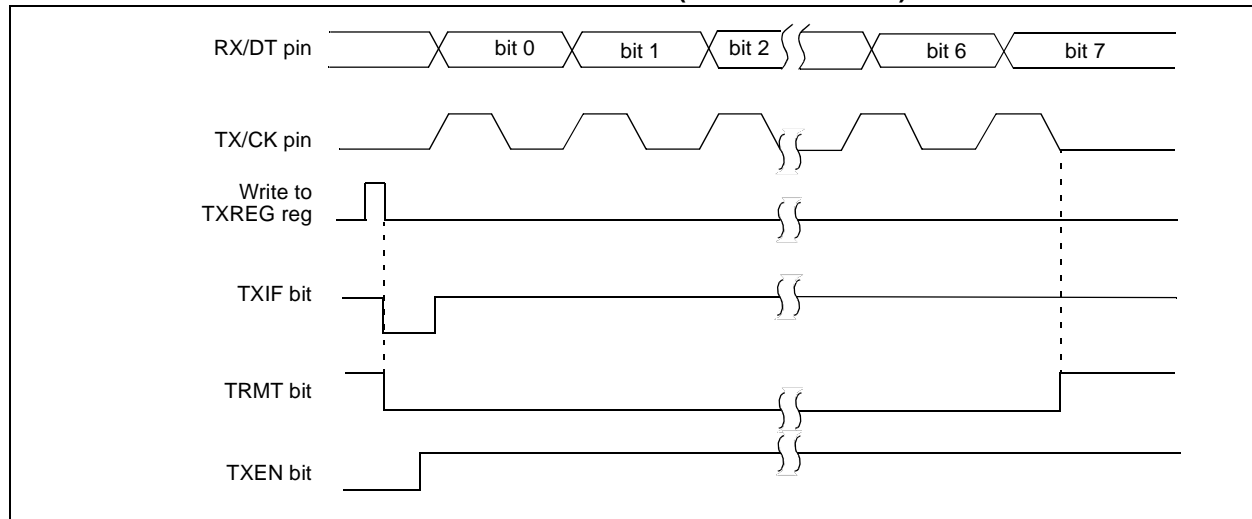


TABLE 16-6: REGISTERS ASSOCIATED WITH SYNCHRONOUS MASTER TRANSMISSION

| Name | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | Value on POR, BOR | Value on all other Resets |
|--------|-------------------------------|--------|--------|--------|--------|--------|--------|--------|-------------------|---------------------------|
| INTCON | GIE | PEIE | T0IE | INTE | RBIE | T0IF | INTF | RBIF | 0000 000x | 0000 000x |
| PIE1 | TMR1GIE | ADIE | RCIE | TXIE | SSPIE | CCP1IE | TMR2IE | TMR1IE | 0000 0000 | 0000 0000 |
| PIR1 | TMR1GIF | ADIF | RCIF | TXIF | SSPIF | CCP1IF | TMR2IF | TMR1IF | 0000 0000 | 0000 0000 |
| RCSTA | SPEN | RX9 | SREN | CREN | ADDEN | FERR | OERR | RX9D | 0000 000x | 0000 000x |
| SPBRG | BRG7 | BRG6 | BRG5 | BRG4 | BRG3 | BRG2 | BRG1 | BRG0 | 0000 0000 | 0000 0000 |
| TRISC | TRISC7 | TRISC6 | TRISC5 | TRISC4 | TRISC3 | TRISC2 | TRISC1 | TRISC0 | 1111 1111 | 1111 1111 |
| TXREG | AUSART Transmit Data Register | | | | | | | | 0000 0000 | 0000 0000 |
| TXSTA | CSRC | TX9 | TXEN | SYNC | — | BRGH | TRMT | TX9D | 0000 -010 | 0000 -010 |

Legend: x = unknown, - = unimplemented read as '0'. Shaded cells are not used for Synchronous Master Transmission.

FIGURE 16-8: SYNCHRONOUS RECEPTION (MASTER MODE, SREN)

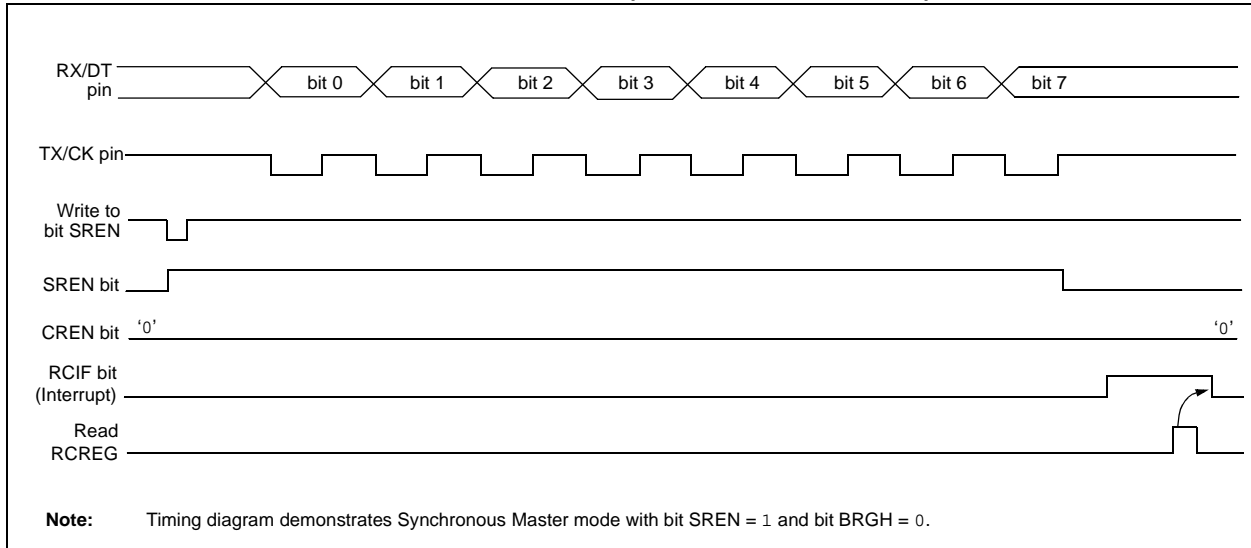


TABLE 16-7: REGISTERS ASSOCIATED WITH SYNCHRONOUS MASTER RECEPTION

| Name | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | Value on POR, BOR | Value on all other Resets |
|--------|------------------------------|--------|--------|--------|--------|--------|--------|--------|-------------------|---------------------------|
| INTCON | GIE | PEIE | T0IE | INTE | RBIE | T0IF | INTF | RBIF | 0000 000x | 0000 000x |
| PIE1 | TMR1GIE | ADIE | RCIE | TXIE | SSPIE | CCP1IE | TMR2IE | TMR1IE | 0000 0000 | 0000 0000 |
| PIR1 | TMR1GIF | ADIF | RCIF | TXIF | SSPIF | CCP1IF | TMR2IF | TMR1IF | 0000 0000 | 0000 0000 |
| RCREG | AUSART Receive Data Register | | | | | | | | 0000 0000 | 0000 0000 |
| RCSTA | SPEN | RX9 | SREN | CREN | ADDEN | FERR | OERR | RX9D | 0000 000x | 0000 000x |
| TRISC | TRISC7 | TRISC6 | TRISC5 | TRISC4 | TRISC3 | TRISC2 | TRISC1 | TRISC0 | 1111 1111 | 1111 1111 |
| TXSTA | CSRC | TX9 | TXEN | SYNC | — | BRGH | TRMT | TX9D | 0000 -010 | 0000 -010 |

Legend: x = unknown, - = unimplemented read as '0'. Shaded cells are not used for Synchronous Master Reception.

16.3.2.3 AUSART Synchronous Slave Reception

The operation of the Synchronous Master and Slave modes is identical (**Section 16.3.1.4 “Synchronous Master Reception”**), with the following exceptions:

- Sleep
- CREN bit is always set, therefore the receiver is never Idle
- SREN bit, which is a “don’t care” in Slave mode

A character may be received while in Sleep mode by setting the CREN bit prior to entering Sleep. Once the word is received, the RSR register will transfer the data to the RCREG register. If the RCIE interrupt enable bit of the PIE1 register is set, the interrupt generated will wake the device from Sleep and execute the next instruction. If the GIE bit is also set, the program will branch to the interrupt vector.

16.3.2.4 Synchronous Slave Reception Setup:

1. Set the SYNC and SPEN bits and clear the CSRC bit.
2. If interrupts are desired, set the RCIE bit of the PIE1 register and the GIE and PEIE bits of the INTCON register.
3. If 9-bit reception is desired, set the RX9 bit.
4. Verify address detection is disabled by clearing the ADDEN bit of the RCSTA register.
5. Set the CREN bit to enable reception.
6. The RCIF bit of the PIR1 register will be set when reception is complete. An interrupt will be generated if the RCIE bit of the PIE1 register was set.
7. If 9-bit mode is enabled, retrieve the Most Significant bit from the RX9D bit of the RCSTA register.
8. Retrieve the eight Least Significant bits from the receive FIFO by reading the RCREG register.
9. If an overrun error occurs, clear the error by either clearing the CREN bit of the RCSTA register.

TABLE 16-9: REGISTERS ASSOCIATED WITH SYNCHRONOUS SLAVE RECEPTION

| Name | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | Value on POR, BOR | Value on all other Resets |
|--------|------------------------------|--------|--------|--------|--------|--------|--------|--------|-------------------|---------------------------|
| INTCON | GIE | PEIE | T0IE | INTE | RBIE | T0IF | INTF | RBIF | 0000 000x | 0000 000x |
| PIE1 | TMR1GIE | ADIE | RCIE | TXIE | SSPIE | CCP1IE | TMR2IE | TMR1IE | 0000 0000 | 0000 0000 |
| PIR1 | TMR1GIF | ADIF | RCIF | TXIF | SSPIF | CCP1IF | TMR2IF | TMR1IF | 0000 0000 | 0000 0000 |
| RCREG | AUSART Receive Data Register | | | | | | | | 0000 0000 | 0000 0000 |
| RCSTA | SPEN | RX9 | SREN | CREN | ADDEN | FERR | OERR | RX9D | 0000 000x | 0000 000x |
| TRISC | TRISC7 | TRISC6 | TRISC5 | TRISC4 | TRISC3 | TRISC2 | TRISC1 | TRISC0 | 1111 1111 | 1111 1111 |
| TXSTA | CSRC | TX9 | TXEN | SYNC | — | BRGH | TRMT | TX9D | 0000 -010 | 0000 -010 |

Legend: x = unknown, – = unimplemented read as ‘0’. Shaded cells are not used for Synchronous Slave Reception.

PIC16(L)F722/3/4/6/7

REGISTER 17-5: SSPMSK: SSP MASK REGISTER

| | | | | | | | |
|-------|-------|-------|-------|-------|-------|-------|-------|
| R/W-1 | R/W-1 | R/W-1 | R/W-1 | R/W-1 | R/W-1 | R/W-1 | R/W-1 |
| MSK7 | MSK6 | MSK5 | MSK4 | MSK3 | MSK2 | MSK1 | MSK0 |
| bit 7 | | | | | | | bit 0 |

Legend:

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

bit 7-1 **MSK<7:1>**: Mask bits
 1 = The received address bit n is compared to SSPADD<n> to detect I²C address match
 0 = The received address bit n is not used to detect I²C address match
 bit 0 **MSK<0>**: Mask bit for I²C Slave Mode, 10-bit Address
 I²C Slave Mode, 10-bit Address (SSPM<3:0> = 0111):
 1 = The received address bit '0' is compared to SSPADD<0> to detect I²C address match
 0 = The received address bit '0' is not used to detect I²C address match
 All other SSP modes: this bit has no effect.

REGISTER 17-6: SSPADD: SSP I²C ADDRESS REGISTER

| | | | | | | | |
|-------|-------|-------|-------|-------|-------|-------|-------|
| R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 |
| ADD7 | ADD6 | ADD5 | ADD4 | ADD3 | ADD2 | ADD1 | ADD0 |
| bit 7 | | | | | | | bit 0 |

Legend:

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

bit 7-0 **ADD<7:0>**: Address bits
 Received address

TABLE 17-7: REGISTERS ASSOCIATED WITH I²C OPERATION

| Name | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | Value on POR, BOR | Value on all other Resets |
|-----------------------|---|--------------------|-------------|--------|--------|-------------|--------|--------|-------------------|---------------------------|
| INTCON | GIE | PEIE | TOIE | INTE | RBIE | TOIF | INTF | RBIF | 0000 000x | 0000 000u |
| PIR1 | TMR1GIF | ADIF | RCIF | TXIF | SSPIF | CCP1IF | TMR2IF | TMR1IF | 0000 0000 | 0000 0000 |
| PIE1 | TMR1GIE | ADIE | RCIE | TXIE | SSPIE | CCP1IE | TMR2IE | TMR1IE | 0000 0000 | 0000 0000 |
| SSPBUF | Synchronous Serial Port Receive Buffer/Transmit Register | | | | | | | | xxxx xxxx | uuuu uuuu |
| SSPADD | Synchronous Serial Port (I ² C mode) Address Register | | | | | | | | 0000 0000 | 0000 0000 |
| SSPCON | WCOL | SSPOV | SSPEN | CKP | SSPM3 | SSPM2 | SSPM1 | SSPM0 | 0000 0000 | 0000 0000 |
| SSPMSK ⁽²⁾ | Synchronous Serial Port (I ² C mode) Address Mask Register | | | | | | | | 1111 1111 | 1111 1111 |
| SSPSTAT | SMP ⁽¹⁾ | CKE ⁽¹⁾ | D \bar{A} | P | S | R \bar{W} | UA | BF | 0000 0000 | 0000 0000 |
| TRISC | TRISC7 | TRISC6 | TRISC5 | TRISC4 | TRISC3 | TRISC2 | TRISC1 | TRISC0 | 1111 1111 | 1111 1111 |

Legend: x = unknown, u = unchanged, - = unimplemented locations read as '0'. Shaded cells are not used by SSP module in I²C mode.

- Note 1:** Maintain these bits clear in I²C mode.
Note 2: Accessible only when SSPM<3:0> = 1001.

18.0 PROGRAM MEMORY READ

The Flash program memory is readable during normal operation over the full VDD range of the device. To read data from Program Memory, five Special Function Registers (SFRs) are used:

- PMCON1
- PMDATL
- PMDATH
- PMADRL
- PMADRH

The value written to the PMADRH:PMADRL register pair determines which program memory location is read. The read operation will be initiated by setting the RD bit of the PMCON1 register. The program memory flash controller takes two instructions to complete the read, causing the second instruction after the setting the RD bit will be ignored. To avoid conflict with program execution, it is recommended that the two instructions following the setting of the RD bit are NOP. When the read completes, the result is placed in the PMDATLH:PMDATL register pair. Refer to Example 18-1 for sample code.

Note: Code-protect does not effect the CPU from performing a read operation on the program memory. For more information, refer to **Section 8.2 “Code Protection”**.

EXAMPLE 18-1: PROGRAM MEMORY READ

Required Sequence

```

BANKSEL PMADRL      ;
MOVWF MS_PROG_ADDR, W ;
MOVWF PMADRH        ;MS Byte of Program Address to read
MOVWF LS_PROG_ADDR, W ;
MOVWF PMADRL        ;LS Byte of Program Address to read
BANKSEL PMCON1      ;
BSF PMCON1, RD      ;Initiate Read
NOP
NOP                  ;Any instructions here are ignored as program
                    ;memory is read in second cycle after BSF

BANKSEL PMDATL      ;
MOVWF PMDATL, W     ;W = LS Byte of Program Memory Read
MOVWF LOWPMBYTE     ;
MOVWF PMDATH, W     ;W = MS Byte of Program Memory Read
MOVWF HIGHPMBYTE    ;
    
```


PIC16(L)F722/3/4/6/7

20.0 IN-CIRCUIT SERIAL PROGRAMMING™ (ICSP™)

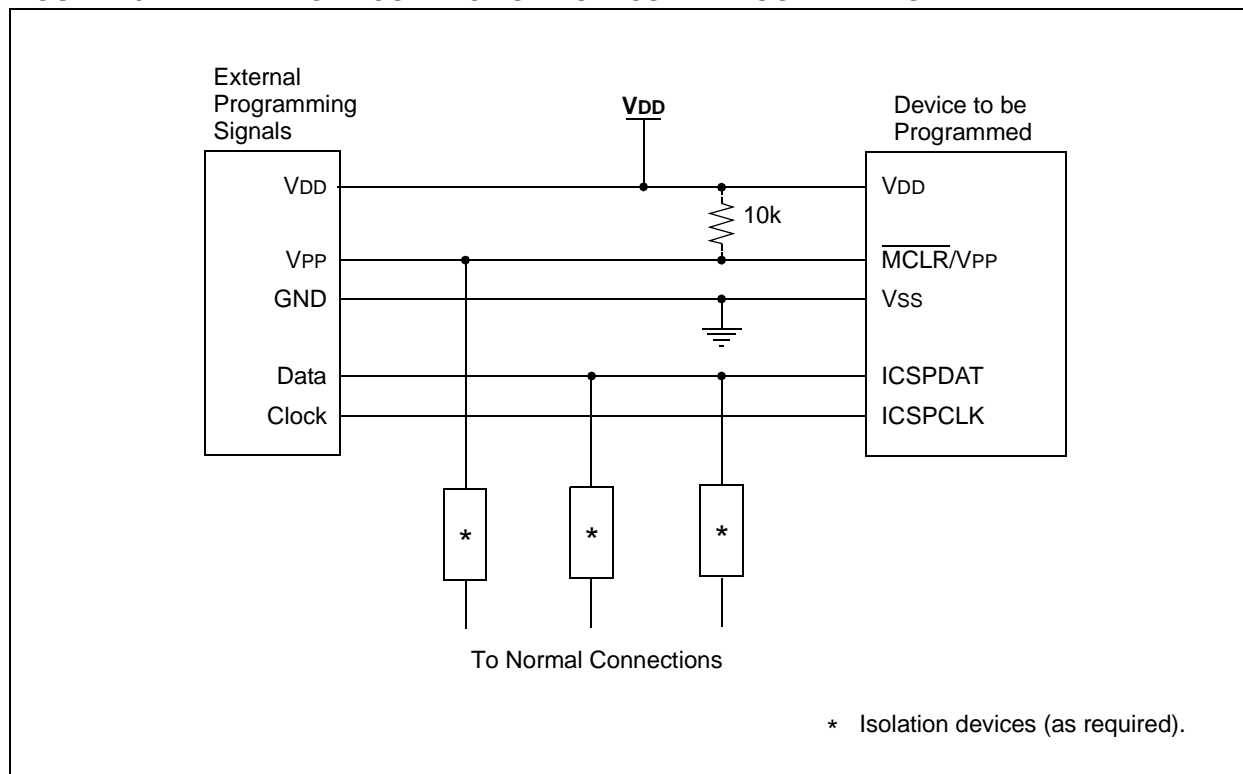
ICSP™ programming allows customers to manufacture circuit boards with unprogrammed devices. Programming can be done after the assembly process allowing the device to be programmed with the most recent firmware or a custom firmware. Five pins are needed for ICSP™ programming:

- ICSPCLK
- ICSPDAT
- MCLR/VPP
- VDD
- VSS

The device is placed into Program/Verify mode by holding the ICSPCLK and ICSPDAT pins low then raising the voltage on $\overline{\text{MCLR/VPP}}$ from 0v to VPP. 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 “PIC16(L)F72x Memory Programming Specification” (DS41332).

Note: The ICD 2 produces a VPP voltage greater than the maximum VPP specification of the PIC16(L)F722/3/4/6/7. When using this programmer, an external circuit, such as the AC164112 MPLAB ICD 2 VPP voltage limiter, is required to keep the VPP voltage within the device specifications.

FIGURE 20-1: TYPICAL CONNECTION FOR ICSP™ PROGRAMMING



DECFSZ Decrement f, Skip if 0

Syntax: [*label*] DECFSZ f,d

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

Operation: $(f) - 1 \rightarrow (\text{destination});$
skip if result = 0

Status Affected: None

Description: The contents of register 'f' are decremented. If 'd' is '0', the result is placed in the W register. If 'd' is '1', the result is placed back in register 'f'.
If the result is '1', the next instruction is executed. If the result is '0', then a NOP is executed instead, making it a 2-cycle instruction.

INCFSZ Increment f, Skip if 0

Syntax: [*label*] INCFSZ f,d

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

Operation: $(f) + 1 \rightarrow (\text{destination});$
skip if result = 0

Status Affected: None

Description: The contents of register 'f' are incremented. If 'd' is '0', the result is placed in the W register. If 'd' is '1', the result is placed back in register 'f'.
If the result is '1', the next instruction is executed. If the result is '0', a NOP is executed instead, making it a 2-cycle instruction.

GOTO Unconditional Branch

Syntax: [*label*] GOTO k

Operands: $0 \leq k \leq 2047$

Operation: $k \rightarrow \text{PC}<10:0>$
 $\text{PCLATH}<4:3> \rightarrow \text{PC}<12:11>$

Status Affected: None

Description: GOTO is an unconditional branch. The 11-bit immediate value is loaded into PC bits <10:0>. The upper bits of PC are loaded from PCLATH<4:3>. GOTO is a 2-cycle instruction.

IORLW Inclusive OR literal with W

Syntax: [*label*] IORLW k

Operands: $0 \leq k \leq 255$

Operation: $(W) .OR. k \rightarrow (W)$

Status Affected: Z

Description: The contents of the W register are OR'ed with the 8-bit literal 'k'. The result is placed in the W register.

INCF Increment f

Syntax: [*label*] INCF f,d

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

Operation: $(f) + 1 \rightarrow (\text{destination})$

Status Affected: Z

Description: The contents of register 'f' are incremented. If 'd' is '0', the result is placed in the W register. If 'd' is '1', the result is placed back in register 'f'.

IORWF Inclusive OR W with f

Syntax: [*label*] IORWF f,d

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

Operation: $(W) .OR. (f) \rightarrow (\text{destination})$

Status Affected: Z

Description: Inclusive OR the W register with register 'f'. If 'd' is '0', the result is placed in the W register. If 'd' is '1', the result is placed back in register 'f'.

PIC16(L)F722/3/4/6/7

23.4 DC Characteristics: PIC16(L)F722/3/4/6/7-I/E (Continued)

| DC CHARACTERISTICS | | | Standard Operating Conditions (unless otherwise stated) Operating temperature $-40^{\circ}\text{C} \leq T_A \leq +85^{\circ}\text{C}$ for industrial $-40^{\circ}\text{C} \leq T_A \leq +125^{\circ}\text{C}$ for extended | | | | |
|---|--------------------|---|--|------|------|-------|--|
| Param No. | Sym. | Characteristic | Min. | Typ† | Max. | Units | Conditions |
| D130 | EP | Cell Endurance | 100 | 1k | — | E/W | Temperature during programming: $10^{\circ}\text{C} \leq T_A \leq 40^{\circ}\text{C}$ |
| D131 | | VDD for Read | V _{MIN} | — | — | V | |
| | | Voltage on $\overline{\text{MCLR}}$ /VPP during Erase/Program | 8.0 | — | 9.0 | V | Temperature during programming: $10^{\circ}\text{C} \leq T_A \leq 40^{\circ}\text{C}$ |
| | | VDD for Bulk Erase | 2.7 | 3 | — | V | Temperature during programming: $10^{\circ}\text{C} \leq T_A \leq 40^{\circ}\text{C}$ |
| D132 | VPEW | VDD for Write or Row Erase | 2.7 | — | — | V | V _{MIN} = Minimum operating voltage V _{MAX} = Maximum operating voltage |
| | I _{PPPGM} | Current on $\overline{\text{MCLR}}$ /VPP during Erase/Write | — | — | 5.0 | mA | Temperature during programming: $10^{\circ}\text{C} \leq T_A \leq 40^{\circ}\text{C}$ |
| | I _{DDPGM} | Current on VDD during Erase/Write | — | — | 5.0 | mA | Temperature during programming: $10^{\circ}\text{C} \leq T_A \leq 40^{\circ}\text{C}$ |
| D133 | TPEW | Erase/Write cycle time | — | — | 2.8 | ms | Temperature during programming: $10^{\circ}\text{C} \leq T_A \leq 40^{\circ}\text{C}$ |
| D134 | TRETD | Characteristic Retention | 40 | — | — | Year | Provided no other specifications are violated |
| V_{CAP} Capacitor Charging | | | | | | | |
| D135 | | Charging current | — | 200 | — | μA | |
| D135A | | Source/sink capability when charging complete | — | 0.0 | — | mA | |

Legend: TBD = To Be Determined

* 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 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 24-27: PIC16F722/3/4/6/7 MAXIMUM BASE I_{PD} vs. V_{DD} , $V_{CAP} = 0.1 \mu F$

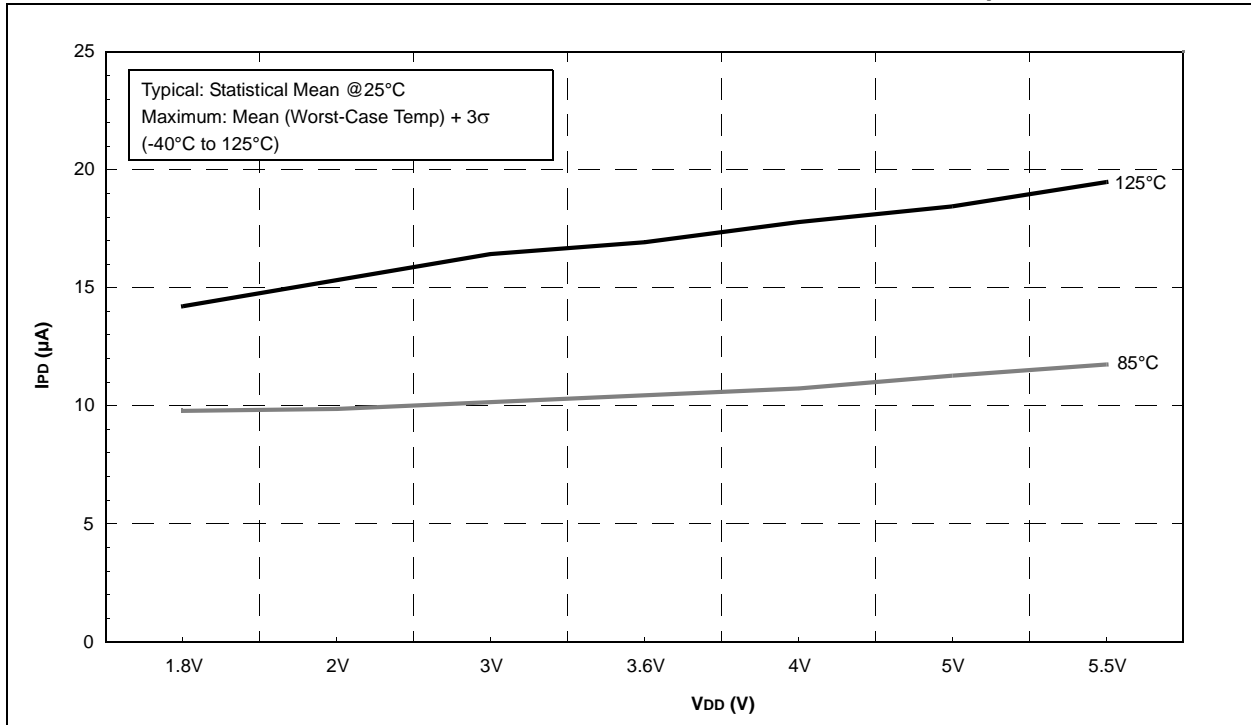
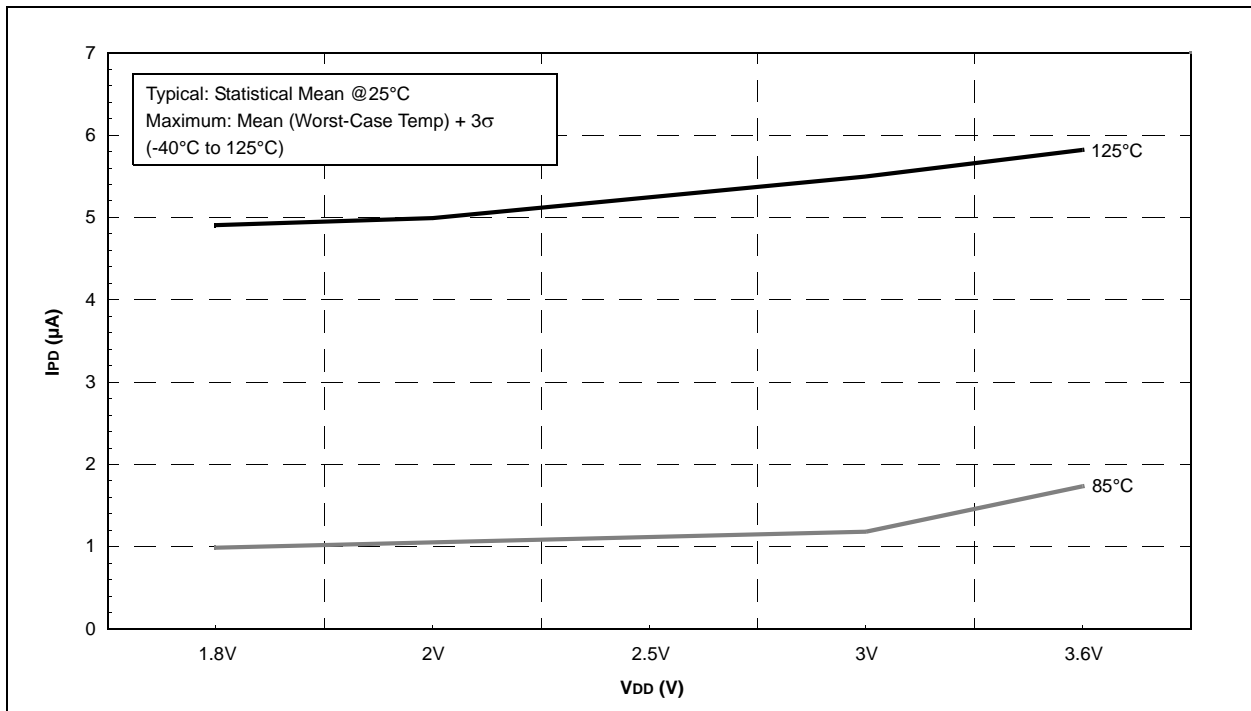


FIGURE 24-28: PIC16LF722/3/4/6/7 MAXIMUM BASE I_{PD} vs. V_{DD}



PIC16(L)F722/3/4/6/7

FIGURE 24-53: V_{OH} vs. I_{OH} OVER TEMPERATURE, $V_{DD} = 3.6V$

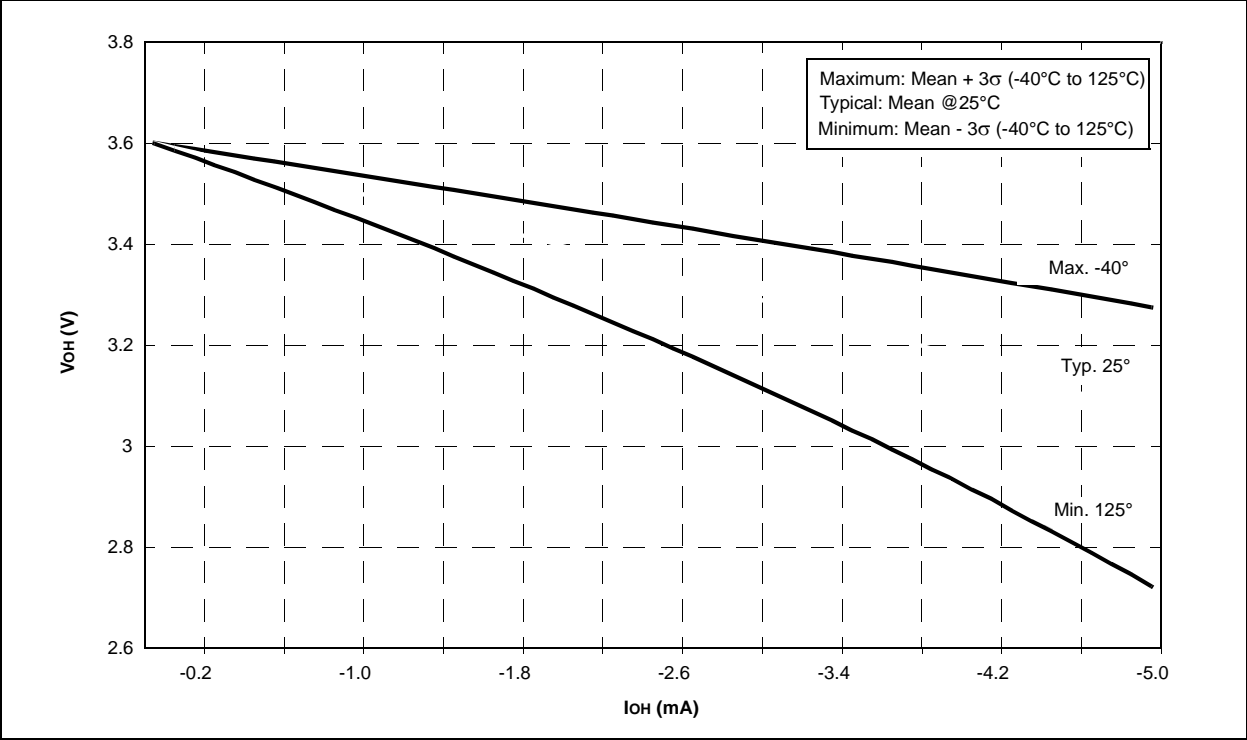


FIGURE 24-54: V_{OH} vs. I_{OH} OVER TEMPERATURE, $V_{DD} = 1.8V$

