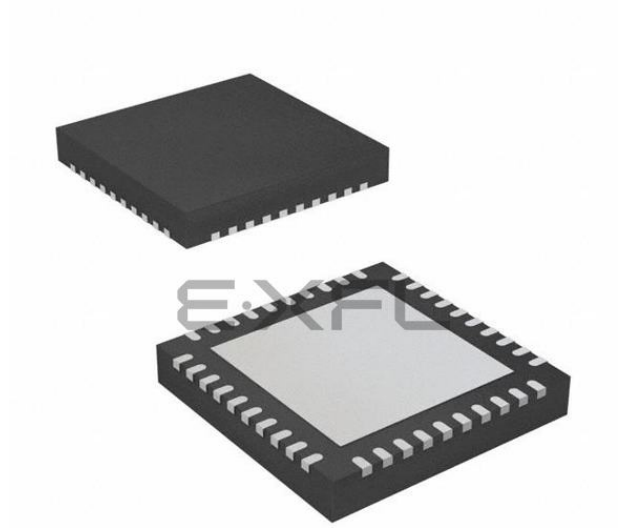


**Welcome to E-XFL.COM**

## 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, 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 | 1K x 8 |
| Voltage - Supply (Vcc/Vdd) | 2.3V ~ 5.5V |
| Data Converters | A/D 35x10b; D/A 1x5b |
| Oscillator Type | Internal |
| Operating Temperature | -40°C ~ 85°C (TA) |
| Mounting Type | Surface Mount |
| Package / Case | 40-UFQFN Exposed Pad |
| Supplier Device Package | 40-UQFN (5x5) |
| Purchase URL | https://www.e-xfl.com/product-detail/microchip-technology/pic16f18875t-i-mv |

**TABLE 1-3: PIC16F18875 PINOUT DESCRIPTION (CONTINUED)**

| Name | Function | Input Type | Output Type | Description |
|---|---|---|---|---|
| RC0/ANC0/T1CKI(1)/T3CKI(1)/T3G(1)/ SMTWIN1(1)/IOCC0/SOSCO | RC0 | TTL/ST | CMOS/OD | General purpose I/O. |
| | ANC0 | AN | — | ADC Channel C0 input. |
| | T1CKI(1) | TTL/ST | — | Timer1 external digital clock input. |
| | T3CKI(1) | TTL/ST | — | Timer3 external digital clock input. |
| | T3G(1) | TTL/ST | — | Timer3 gate input. |
| | SMTWIN1(1) | TTL/ST | — | Signal Measurement Timer1 (SMT1) input. |
| | IOCC0 | TTL/ST | — | Interrupt-on-change input. |
| | SOSCO | — | AN | 32.768 kHz secondary oscillator crystal driver output. |
| RC1/ANC1/SMTSIG1(1)/CCP2(1)/ IOCC1/SOSCI | RC1 | TTL/ST | CMOS/OD | General purpose I/O. |
| | ANC1 | AN | — | ADC Channel C1 input. |
| | SMTSIG1(1) | TTL/ST | — | Signal Measurement Timer1 (SMT1) signal input. |
| | CCP2(1) | TTL/ST | CMOS/OD | Capture/compare/PWM2 (default input location for capture function). |
| | IOCC1 | TTL/ST | — | Interrupt-on-change input. |
| | SOSCI | AN | — | 32.768 kHz secondary oscillator crystal driver input. |
| RC2/ANC2/T5CKI(1)/CCP1(1)/IOCC2 | RC2 | TTL/ST | CMOS/OD | General purpose I/O. |
| | ANC2 | AN | — | ADC Channel C2 input. |
| | T5CKI(1) | TTL/ST | — | Timer5 external digital clock input. |
| | CCP1(1) | TTL/ST | CMOS/OD | Capture/compare/PWM1 (default input location for capture function). |
| | IOCC2 | TTL/ST | — | Interrupt-on-change input. |
| RC3/ANC3/SCL1(3,4)/SCK1(1)/ T2IN(1)/IOCC3 | RC3 | TTL/ST | CMOS/OD | General purpose I/O. |
| | ANC3 | AN | — | ADC Channel C3 input. |
| | SCL1(3,4) | $I^2C$/SMBus | OD | MSSP1 $I^2C$ clock input/output. |
| | SCK1(1) | TTL/ST | CMOS/OD | MSSP1 SPI clock input/output (default input location, SCK1 is a PPS remappable input and output). |
| | T2IN(1) | TTL/ST | — | Timer2 external input. |
| | IOCC3 | TTL/ST | — | Interrupt-on-change input. |
| RC4/ANC4/SDA1(3,4)/SDI1(1)/IOCC4 | RC4 | TTL/ST | CMOS/OD | General purpose I/O. |
| | ANC4 | AN | — | ADC Channel C4 input. |
| | SDA1(3,4) | $I^2C$/SMBus | OD | MSSP1 $I^2C$ serial data input/output. |
| | SDI1(1) | TTL/ST | — | MSSP1 SPI serial data input. |
| | IOCC4 | TTL/ST | — | Interrupt-on-change input. |
| RC5/ANC5/T4IN(1)/IOCC5 | RC5 | TTL/ST | CMOS/OD | General purpose I/O. |
| | ANC5 | AN | — | ADC Channel C5 input. |
| | T4IN(1) | TTL/ST | — | Timer4 external input. |
| | IOCC5 | TTL/ST | — | Interrupt-on-change input. |

**Legend:** AN = Analog input or output CMOS = CMOS compatible input or output OD = Open-Drain
TTL = TTL compatible input ST = Schmitt Trigger input with CMOS levels $I^2C$ = Schmitt Trigger input with $I^2C$HV= High Voltage XTAL= Crystal levels

**Note 1:** This is a PPS remappable input signal. The input function may be moved from the default location shown to one of several other PORTx pins. Refer to Table 13-1 for details on which PORT pins may be used for this signal.

**2:** All output signals shown in this row are PPS remappable. These signals may be mapped to output onto one of several PORTx pin options as described in Table 13-3.

**3:** This is a bidirectional signal. For normal module operation, the firmware should map this signal to the same pin in both the PPS input and PPS output registers.

**4:** These pins are configured for $I^2C$ logic levels. The SCLx/SDAx signals may be assigned to any of the RB1/RB2/RC3/RC4 pins. PPS assignments to the other pins (e.g., RA5) will operate, but input logic levels will be standard TTL/ST, as selected by the INLVL register, instead of the $I^2C$ specific or SMBus input buffer thresholds.

**TABLE 3-13: SPECIAL FUNCTION REGISTER SUMMARY BANKS 0-31 (CONTINUED)**

| Address | Name | PIC16(L)F18855 | PIC16(L)F18875 | 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 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **Bank 4** | | | | | | | | | | | | | |
| **CPU CORE REGISTERS; see Table 3-2 for specifics** | | | | | | | | | | | | | |
| 20Ch | TMR1L | | | Holding Register for the Least Significant Byte of the 16-bit TMR1 Register | | | | | | | | 0000 0000 | uuuu uuuu |
| 20Dh | TMR1H | | | Holding Register for the Most Significant Byte of the 16-bit TMR1 Register | | | | | | | | 0000 0000 | uuuu uuuu |
| 20Eh | T1CON | | | — | — | CKPS<1:0> | | — | $\overline{\text{SYNC}}$ | RD16 | ON | --00 -000 | --uu -uuu |
| 20Fh | T1GCON | | | GE | GPOL | GTM | GSPM | GGO/$\overline{\text{DONE}}$ | GVAL | — | — | 0000 0x-- | uuuu ux-- |
| 210h | T1GATE | | | — | — | — | GSS<4:0> | | | | | ---0 0000 | ---u uuuu |
| 211h | T1CLK | | | — | — | — | — | CS<3:0> | | | | ---- 0000 | ---- uuuu |
| 212h | TMR3L | | | Holding Register for the Least Significant Byte of the 16-bit TMR3 Register | | | | | | | | 0000 0000 | uuuu uuuu |
| 213h | TMR3H | | | Holding Register for the Most Significant Byte of the 16-bit TMR3 Register | | | | | | | | 0000 0000 | uuuu uuuu |
| 214h | T3CON | | | — | — | CKPS<1:0> | | — | $\overline{\text{SYNC}}$ | RD16 | ON | --00 -000 | --uu -uuu |
| 215h | T3GCON | | | GE | GPOL | GTM | GSPM | GGO/$\overline{\text{DONE}}$ | GVAL | — | — | 0000 0x-- | uuuu ux-- |
| 216h | T3GATE | | | — | — | — | GSS<4:0> | | | | | ---0 0000 | ---u uuuu |
| 217h | T3CLK | | | — | — | — | — | CS<3:0> | | | | ---- 0000 | ---- uuuu |
| 218h | TMR5L | | | Holding Register for the Least Significant Byte of the 16-bit TMR5 Register | | | | | | | | 0000 0000 | uuuu uuuu |
| 219h | TMR5H | | | Holding Register for the Most Significant Byte of the 16-bit TMR5 Register | | | | | | | | 0000 0000 | uuuu uuuu |
| 21Ah | T5CON | | | — | — | CKPS<1:0> | | — | $\overline{\text{SYNC}}$ | RD16 | ON | --00 -000 | --uu -uuu |
| 21Bh | T5GCON | | | GE | GPOL | GTM | GSPM | GGO/$\overline{\text{DONE}}$ | GVAL | — | — | 0000 0x-- | uuuu ux-- |
| 21Ch | T5GATE | | | — | — | — | GSS<4:0> | | | | | ---0 0000 | ---u uuuu |
| 21Dh | T5CLK | | | — | — | — | — | CS<3:0> | | | | ---- 0000 | ---- uuuu |
| 21Eh | CCPTMRS0 | | | C4TSEL<1:0> | | C3TSEL<1:0> | | C2TSEL<1:0> | | C1TSEL<1:0> | | 0101 0101 | 0101 0101 |
| 21Fh | CCPTMRS1 | | | — | — | P7TSEL<1:0> | | P6TSEL<1:0> | | C5TSEL<1:0> | | --01 0101 | --01 0101 |

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

**Note 1:** Register present on PIC16F18855/75 devices only.

**2:** Unimplemented, read as '1'.

**TABLE 3-13: SPECIAL FUNCTION REGISTER SUMMARY BANKS 0-31 (CONTINUED)**

| Address | Name | PIC16(L)F18855 | PIC16(L)F18875 | 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 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **Banks 16** | | | | | | | | | | | | | |
| **CPU CORE REGISTERS; see Table 3-2 for specifics** | | | | | | | | | | | | | |
| 80Ch | WDTCON0 | | | — | — | PS<4:0> | | | | | SEN | --qq qqq0 | --qq qqq0 |
| 80Dh | WDTCON1 | | | — | WDTCS<2:0> | | | — | WINDOW<2:0> | | | -qqq -qqq | -qqq -qqq |
| 80Eh | WDTPSL | | | PSCNT<7:0> | | | | | | | | 0000 0000 | 0000 0000 |
| 80Fh | WDTPSH | | | PSCNT<7:0> | | | | | | | | 0000 0000 | 0000 0000 |
| 810h | WDTTMR | | | — | WDTTMR<3:0> | | | | STATE | PSCNT<17:16> | | -000 0000 | -000 0000 |
| 811h | BORCON | | | SBOREN | — | — | — | — | — | — | BORRDY | 1--- ---q | u--- ---u |
| 812h | VREGCON(1) | | | — | — | — | — | — | — | VREGPM | Reserved | ---- --01 | ---- --01 |
| 813h | PCON0 | | | STKOVF | STKUNF | $\overline{\text{WDTWV}}$ | $\overline{\text{RWDT}}$ | $\overline{\text{RMCLR}}$ | $\overline{\text{RI}}$ | $\overline{\text{POR}}$ | $\overline{\text{BOR}}$ | 0011 11qq | qqqq qquu |
| 814h | CCDCON | | | CCDEN | — | — | — | — | — | CCDS<1:0> | | 0--- --xx | 0--- --uu |
| 815h | — | — | | Unimplemented | | | | | | | | — | — |
| 816h | — | — | | Unimplemented | | | | | | | | — | — |
| 817h | — | — | | Unimplemented | | | | | | | | — | — |
| 818h | — | — | | Unimplemented | | | | | | | | — | — |
| 819h | — | — | | Unimplemented | | | | | | | | — | — |
| 81Ah | NVMADRL | | | NVMADR<7:0> | | | | | | | | 0000 0000 | 0000 0000 |
| 81Bh | NVMADRH | | | —(2) | NVMADR<14:8> | | | | | | | 1000 0000 | 1000 0000 |
| 81Ch | NVMDATL | | | NVMDAT<7:0> | | | | | | | | 0000 0000 | 0000 0000 |
| 81Dh | NVMDATH | | | — | — | NVMDAT<13:8> | | | | | | --00 0000 | --00 0000 |
| 81Eh | NVMCON1 | | | — | NVMREGS | LWLO | FREE | WRERR | WREN | WR | RD | -000 x000 | -000 q000 |
| 81Fh | NVMCON2 | | | NVMCON2<7:0> | | | | | | | | 0000 0000 | 0000 0000 |

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

**Note 1:** Register present on PIC16F18855/75 devices only.

**2:** Unimplemented, read as '1'.

**TABLE 3-13: SPECIAL FUNCTION REGISTER SUMMARY BANKS 0-31 (CONTINUED)**

| Address | Name | PIC16(L)F18855 | PIC16(L)F18875 | 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 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **Bank 30 (Continued)** | | | | | | | | | | | | | |
| F24h | RC4PPS | | | — | — | RC4PPS<5:0> | | | | | | --00 0000 | --uu uuuu |
| F25h | RC5PPS | | | — | — | RC5PPS<5:0> | | | | | | --00 0000 | --uu uuuu |
| F26h | RC6PPS | | | — | — | RC6PPS<5:0> | | | | | | --00 0000 | --uu uuuu |
| F27h | RC7PPS | | | — | — | RC7PPS<5:0> | | | | | | --00 0000 | --uu uuuu |
| F28h | RD0PPS | — | X | — | — | RD0PPS<5:0> | | | | | | --00 0000 | --uu uuuu |
| | | X | — | Unimplemented | | | | | | | | ---- ---- | ---- ---- |
| F29h | RD1PPS | — | X | — | — | RD1PPS<5:0> | | | | | | --00 0000 | --uu uuuu |
| | | X | — | Unimplemented | | | | | | | | ---- ---- | ---- ---- |
| F2Ah | RD2PPS | — | X | — | — | RD2PPS<5:0> | | | | | | --00 0000 | --uu uuuu |
| | | X | — | Unimplemented | | | | | | | | ---- ---- | ---- ---- |
| F2Bh | RD3PPS | — | X | — | — | RD3PPS<5:0> | | | | | | --00 0000 | --uu uuuu |
| | | X | — | Unimplemented | | | | | | | | ---- ---- | ---- ---- |
| F2Ch | RD4PPS | — | X | — | — | RD4PPS<5:0> | | | | | | --00 0000 | --uu uuuu |
| | | X | — | Unimplemented | | | | | | | | ---- ---- | ---- ---- |
| F2Dh | RD5PPS | — | X | — | — | RD5PPS<5:0> | | | | | | --00 0000 | --uu uuuu |
| | | X | — | Unimplemented | | | | | | | | ---- ---- | ---- ---- |
| F2Eh | RD6PPS | — | X | — | — | RD6PPS<5:0> | | | | | | --00 0000 | --uu uuuu |
| | | X | — | Unimplemented | | | | | | | | ---- ---- | ---- ---- |
| F2Fh | RD7PPS | — | X | — | — | RD7PPS<5:0> | | | | | | --00 0000 | --uu uuuu |
| | | X | — | Unimplemented | | | | | | | | ---- ---- | ---- ---- |
| F30h | RE0PPS | — | X | — | — | RE0PPS<5:0> | | | | | | --00 0000 | --uu uuuu |
| | | X | — | Unimplemented | | | | | | | | ---- ---- | ---- ---- |

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

**Note 1:** Register present on PIC16F18855/75 devices only.

**2:** Unimplemented, read as '1'.

#### 6.2.1.4 4x PLL

The oscillator module contains a PLL that can be used with external clock sources to provide a system clock source. The input frequency for the PLL must fall within specifications. See the PLL Clock Timing Specifications in Table 37-9.

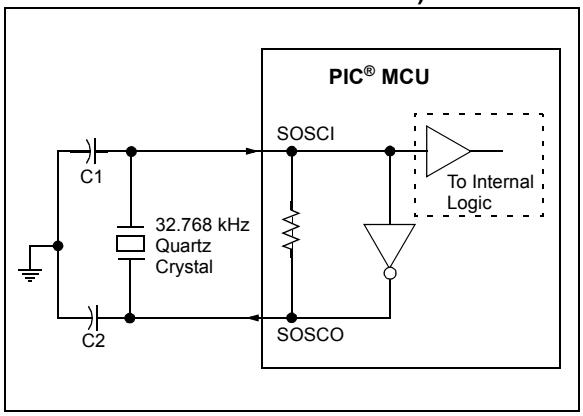
The PLL may be enabled for use by one of two methods:

1. Program the RSTOSC bits in the Configuration Word 1 to enable the EXTOSC with 4x PLL.
2. Write the NOSC bits in the OSCCON1 register to enable the EXTOSC with 4x PLL.

#### 6.2.1.5 Secondary Oscillator

The secondary oscillator is a separate oscillator block that can be used as an alternate system clock source. The secondary oscillator is optimized for 31 kHz, and can be used with an external crystal oscillator connected to the SOSCI and SOSCO device pins, or an external clock source connected to the SOSCIN pin. The secondary oscillator can be selected during run-time using clock switching. Refer to **Section 6.3 "Clock Switching"** for more information.

**FIGURE 6-5: QUARTZ CRYSTAL OPERATION (SECONDARY OSCILLATOR)**




**Note 1:** Quartz crystal characteristics vary according to type, package and manufacturer. The user should consult the manufacturer data sheets for specifications and recommended application.

**2:** Always verify oscillator performance over the $V_{DD}$ and temperature range that is expected for the application.

**3:** For oscillator design assistance, reference the following Microchip Application Notes:

- AN826, "*Crystal Oscillator Basics and Crystal Selection for rfPIC® and PIC® Devices*" (DS00826)
- AN849, "*Basic PIC® Oscillator Design*" (DS00849)
- AN943, "*Practical PIC® Oscillator Analysis and Design*" (DS00943)
- AN949, "*Making Your Oscillator Work*" (DS00949)
- TB097, "*Interfacing a Micro Crystal MS1V-T1K 32.768 kHz Tuning Fork Crystal to a PIC16F690/SS*" (DS91097)
- AN1288, "*Design Practices for Low-Power External Oscillators*" (DS01288)

**REGISTER 6-4: OSCSTAT: OSCILLATOR STATUS REGISTER 1**

| R-q/q | R-0/q | R-0/q | R-0/q | R-q/q | R-q/q | U-0 | R-q/q |
|---|---|---|---|---|---|---|---|
| EXTOR | HFOR | MFOR | LFOR | SOR | ADOR | — | PLLR |
| 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 **EXTOR:** EXTOSC (external) Oscillator Ready bit
1 = The oscillator is ready to be used
0 = The oscillator is not enabled, or is not yet ready to be used.

bit 6 **HFOR:** HFINTOSC Oscillator Ready bit
1 = The oscillator is ready to be used
0 = The oscillator is not enabled, or is not yet ready to be used.

bit 5 **MFOR:** MFINTOSC Oscillator Ready bit
1 = The oscillator is ready to be used
0 = The oscillator is not enabled, or is not yet ready to be used

bit 4 **LFOR:** LFINTOSC Oscillator Ready bit
1 = The oscillator is ready to be used
0 = The oscillator is not enabled, or is not yet ready to be used.

bit 3 **SOR:** Secondary (Timer1) Oscillator Ready bit
1 = The oscillator is ready to be used
0 = The oscillator is not enabled, or is not yet ready to be used.

bit 2 **ADOR:** CRC Oscillator Ready bit
1 = The oscillator is ready to be used
0 = The oscillator is not enabled, or is not yet ready to be used

bit 1 **Unimplemented:** Read as ‘0’

bit 0 **PLLR:** PLL is Ready bit
1 = The PLL is ready to be used
0 = The PLL is not enabled, the required input source is not ready, or the PLL is not locked.

**REGISTER 6-7: OSCTUNE: HFINTOSC TUNING REGISTER**

| U-0 | U-0 | R/W-1/1 | R/W-0/0 | R/W-0/0 | R/W-0/0 | R/W-0/0 | R/W-0/0 |
|---|---|---|---|---|---|---|---|
| — | — | HFTUN<5:0> | | | | | |
| bit 7 | | | | | | | bit 0 |

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

bit 7-6 **Unimplemented:** Read as ‘0’.

bit 5-0 **HFTUN<5:0>:** HFINTOSC Frequency Tuning bits

11 1111 = Maximum frequency
•
•
•
10 0001
10 0000 = Center frequency. Oscillator module is running at the calibrated frequency (default value).
01 1111
•
•
•
00 0000 = Minimum frequency.

## 7.1 Operation

Interrupts are disabled upon any device Reset. They are enabled by setting the following bits:

- GIE bit of the INTCON register
- Interrupt Enable bit(s) for the specific interrupt event(s)
- PEIE bit of the INTCON register (if the Interrupt Enable bit of the interrupt event is contained in the PIEx registers)

The PIR1, PIR2, PIR3 and PIR4 registers record individual interrupts via interrupt flag bits. Interrupt flag bits will be set, regardless of the status of the GIE, PEIE and individual interrupt enable bits.

The following events happen when an interrupt event occurs while the GIE bit is set:

- Current prefetched instruction is flushed
- GIE bit is cleared
- Current Program Counter (PC) is pushed onto the stack
- Critical registers are automatically saved to the shadow registers (See **"Section 7.5 "Automatic Context Saving"**)
- PC is loaded with the interrupt vector 0004h

The firmware within the Interrupt Service Routine (ISR) should determine the source of the interrupt by polling the interrupt flag bits. The interrupt flag bits must be cleared before exiting the ISR to avoid repeated interrupts. Because the GIE bit is cleared, any interrupt that occurs while executing the ISR will be recorded through its interrupt flag, but will not cause the processor to redirect to the interrupt vector.

The `RETFIE` instruction exits the ISR by popping the previous address from the stack, restoring the saved context from the shadow registers and setting the GIE bit.

For additional information on a specific interrupt's operation, refer to its peripheral chapter.

| | |
|---|---|
| **Note 1:** | Individual interrupt flag bits are set, regardless of the state of any other enable bits. |
| **2:** | All interrupts will be ignored while the GIE bit is cleared. Any interrupt occurring while the GIE bit is clear will be serviced when the GIE bit is set again. |

## 7.2 Interrupt Latency

Interrupt latency is defined as the time from when the interrupt event occurs to the time code execution at the interrupt vector begins. The latency for synchronous interrupts is three or four instruction cycles. For asynchronous interrupts, the latency is three to five instruction cycles, depending on when the interrupt occurs. See Figure 7-2 and Figure 7-3 for more details.

**REGISTER 7-11: 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:** TMR0 Overflow Interrupt Flag bit
- 1 = TMR0 register has overflowed (must be cleared in software)
- 0 = TMR0 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 13-1).

**2:** The IOCIF bits are the logical OR of all the IOCAF-IOCEF flags. Therefore, to clear the IOCIF flag, application firmware should 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.

**TABLE 7-1: SUMMARY OF REGISTERS ASSOCIATED WITH INTERRUPTS**

| Name | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | Register on Page |
|---|---|---|---|---|---|---|---|---|---|
| INTCON | GIE | PEIE | — | — | — | — | — | INTEDG | 134 |
| PIE0 | — | — | TMR0IE | IOCIE | — | — | — | INTE | 135 |
| PIE1 | OSFIE | CSWIE | — | — | — | — | ADTIE | ADIE | 136 |
| PIE2 | — | ZCDIE | — | — | — | — | C2IE | C1IE | 137 |
| PIE3 | — | — | RCIE | TXIE | BCL2IE | SSP2IE | BCL1IE | SSP1IE | 138 |
| PIE4 | — | — | TMR6IE | TMR5IE | TMR4IE | TMR3IE | TMR2IE | TMR1IE | 139 |
| PIE5 | CLC4IE | CLC3IE | CLC2IE | CLC1IE | — | TMR5GIE | TMR3GIE | TMR1GIE | 140 |
| PIE6 | — | — | — | CCP5IE | CCP4IE | CCP3IE | CCP2IE | CCP1IE | 141 |
| PIE7 | SCANIE | CRCIE | NVMIE | NCO1IE | — | CWG3IE | CWG2IE | CWG1IE | 142 |
| PIE8 | — | — | SMT2PWAIE | SMT2PRAIE | SMT2IE | SMT1PWAIE | SMT1PRAIE | SMT1IE | 143 |
| PIR0 | — | — | TMR0IF | IOCIF | — | — | — | INTF | 144 |
| PIR1 | OSFIF | CSWIF | — | — | — | — | ADTIF | ADIF | 145 |
| PIR2 | — | ZCDIF | — | — | — | — | C2IF | C1IF | 146 |
| PIR3 | — | — | RCIF | TXIF | BCL2IF | SSP2IF | BCL1IF | SSP1IF | 147 |
| PIR4 | — | — | TMR6IF | TMR5IF | TMR4IF | TMR3IF | TMR2IF | TMR1IF | 148 |
| PIR5 | CLC4IF | CLC3IF | CLC2IF | CLC1IF | — | TMR5GIF | TMR3GIF | TMR1GIF | 149 |
| PIR6 | — | — | — | CCP5IF | CCP4IF | CCP3IF | CCP2IF | CCP1IF | 150 |
| PIR7 | SCANIF | CRCIF | NVMIF | NCO1IF | — | CWG3IF | CWG2IF | CWG1IF | 152 |
| PIR8 | — | — | SMT2PWAIF | SMT2PRAIF | SMT2IF | SMT1PWAIF | SMT1PRAIF | SMT1IF | 153 |

**Legend:** — = unimplemented location, read as '0'. Shaded cells are not used by interrupts.

## 10.5 Register Definitions: Flash Program Memory Control

### REGISTER 10-1: NVMDATL: NONVOLATILE MEMORY DATA LOW BYTE REGISTER

| R/W-x/u | R/W-x/u | R/W-x/u | R/W-x/u | R/W-x/u | R/W-x/u | R/W-x/u | R/W-x/u |
|---|---|---|---|---|---|---|---|
| NVMDAT<7:0> | | | | | | | |
| bit 7 | | | | | | | bit 0 |

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

bit 7-0 **NVMDAT<7:0>**: Read/write value for Least Significant bits of program memory

### REGISTER 10-2: NVMDATH: NONVOLATILE MEMORY DATA HIGH BYTE REGISTER

| U-0 | U-0 | R/W-x/u | R/W-x/u | R/W-x/u | R/W-x/u | R/W-x/u | R/W-x/u |
|---|---|---|---|---|---|---|---|
| — | — | NVMDAT<13:8> | | | | | |
| bit 7 | | | | | | | bit 0 |

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

bit 7-6 **Unimplemented:** Read as ‘0’

bit 5-0 **NVMDAT<13:8>**: Read/write value for Most Significant bits of program memory

### REGISTER 10-3: NVMADRL: NONVOLATILE MEMORY ADDRESS LOW BYTE 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 |
|---|---|---|---|---|---|---|---|
| NVMADR<7:0> | | | | | | | |
| bit 7 | | | | | | | bit 0 |

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

bit 7-0 **NVMADR<7:0>**: Specifies the Least Significant bits for program memory address

### REGISTER 10-4: NVMADRH: NONVOLATILE MEMORY ADDRESS HIGH BYTE REGISTER

| U-1 | 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 |
|---|---|---|---|---|---|---|---|
| —(1) | NVMADR<14:8> | | | | | | |
| 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 **Unimplemented:** Read as ‘1’

bit 6-0 **NVMADR<14:8>**: Specifies the Most Significant bits for program memory address

**Note 1:** Bit is undefined while WR = 1 (during the EEPROM write operation it may be ‘0’ or ‘1’).

## 12.4 PORTA Registers

### 12.4.1 DATA REGISTER

PORTA is an 8-bit wide, bidirectional port. The corresponding data direction register is TRISA (Register 12-3). Setting a TRISA bit (= 1) will make the corresponding PORTA pin an input (i.e., disable the output driver). Clearing a TRISA bit (= 0) will make the corresponding PORTA pin an output (i.e., enables output driver and puts the contents of the output latch on the selected pin). Example 12.4.9 shows how to initialize PORTA.

Reading the PORTA register (Register 12-2) reads the status of the pins, whereas writing to it will write to the PORT latch. All write operations are read-modify-write operations. Therefore, a write to a port implies that the port pins are read, this value is modified and then written to the PORT data latch (LATA).

The PORT data latch LATA (Register 12-4) holds the output port data, and contains the latest value of a LATA or PORTA write.

**EXAMPLE 12-1: INITIALIZING PORTA**

```
; This code example illustrates
; initializing the PORTA register. The
; other ports are initialized in the same
; manner.

BANKSEL  PORTA        ;
CLRF     PORTA        ;Init PORTA
BANKSEL  LATA         ;Data Latch
CLRF     LATA         ;
BANKSEL  ANSELA       ;
CLRF     ANSELA       ;digital I/O
BANKSEL  TRISA        ;
MOVLW    B'00111000'  ;Set RA<5:3> as inputs
MOVWF    TRISA        ;and set RA<2:0> as
                      ;outputs
```

### 12.4.2 DIRECTION CONTROL

The TRISA register (Register 12-3) controls the PORTA pin output drivers, even when they are being used as analog inputs. The user should ensure the bits in the TRISA register are maintained set when using them as analog inputs. I/O pins configured as analog inputs always read '0'.

### 12.4.3 OPEN-DRAIN CONTROL

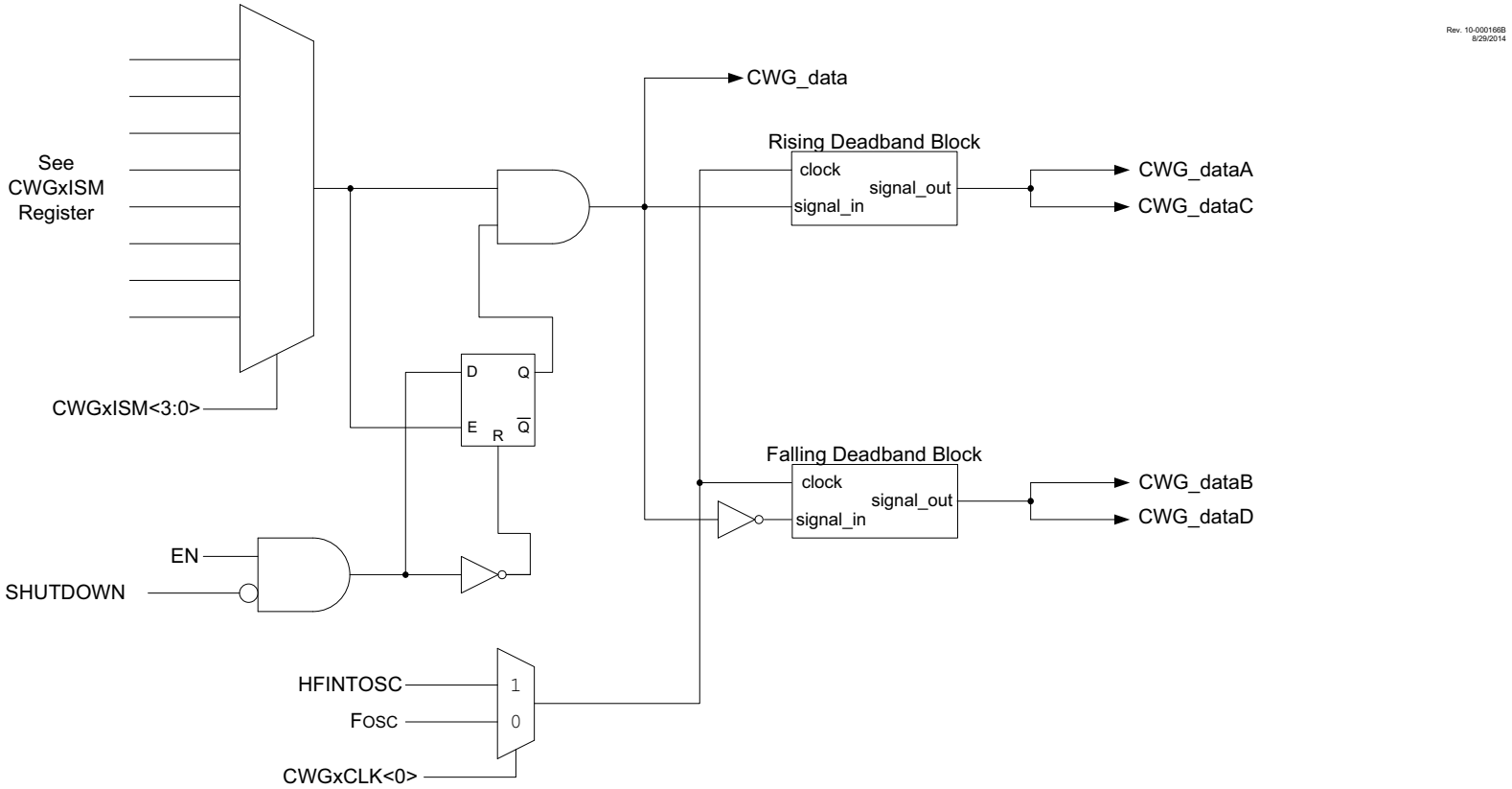
The ODCONA register (Register 12-7) controls the open-drain feature of the port. Open-drain operation is independently selected for each pin. When an ODCONA bit is set, the corresponding port output becomes an open-drain driver capable of sinking current only. When an ODCONA bit is cleared, the corresponding port output pin is the standard push-pull drive capable of sourcing and sinking current.

**Note:** It is not necessary to set open-drain control when using the pin for $I^2C$; the $I^2C$ module controls the pin and makes the pin open-drain.

### 12.4.4 SLEW RATE CONTROL

The SLRCONA register (Register 12-8) controls the slew rate option for each port pin. Slew rate control is independently selectable for each port pin. When an SLRCONA bit is set, the corresponding port pin drive is slew rate limited. When an SLRCONA bit is cleared, The corresponding port pin drive slews at the maximum rate possible.

**FIGURE 20-1: SIMPLIFIED CWG BLOCK DIAGRAM (HALF-BRIDGE MODE)**



## 21.0 ZERO-CROSS DETECTION (ZCD) MODULE

The ZCD module detects when an A/C signal crosses through the ground potential. The actual zero crossing threshold is the zero crossing reference voltage, VCPINV, which is typically 0.75V above ground.

The connection to the signal to be detected is through a series current limiting resistor. The module applies a current source or sink to the ZCD pin to maintain a constant voltage on the pin, thereby preventing the pin voltage from forward biasing the ESD protection diodes. When the applied voltage is greater than the reference voltage, the module sinks current. When the applied voltage is less than the reference voltage, the module sources current. The current source and sink action keeps the pin voltage constant over the full range of the applied voltage. The ZCD module is shown in the simplified block diagram Figure 21-2.

The ZCD module is useful when monitoring an A/C waveform for, but not limited to, the following purposes:

- A/C period measurement
- Accurate long term time measurement
- Dimmer phase delayed drive
- Low EMI cycle switching

### 21.1 External Resistor Selection

The ZCD module requires a current limiting resistor in series with the external voltage source. The impedance and rating of this resistor depends on the external source peak voltage. Select a resistor value that will drop all of the peak voltage when the current through the resistor is nominally 300 μA. Refer to Equation 21-1 and Figure 21-1. Make sure that the ZCD I/O pin internal weak pull-up is disabled so it does not interfere with the current source and sink.

**EQUATION 21-1: EXTERNAL RESISTOR**

$$R_{SERIES} = \frac{V_{PEAK}}{3 \times 10^{-4}}$$

**FIGURE 21-1: EXTERNAL VOLTAGE**

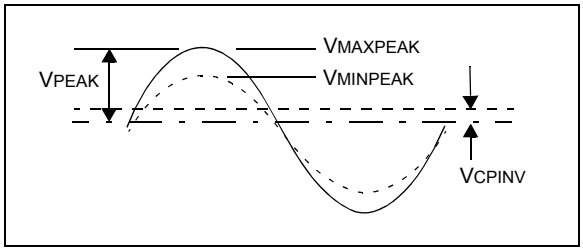
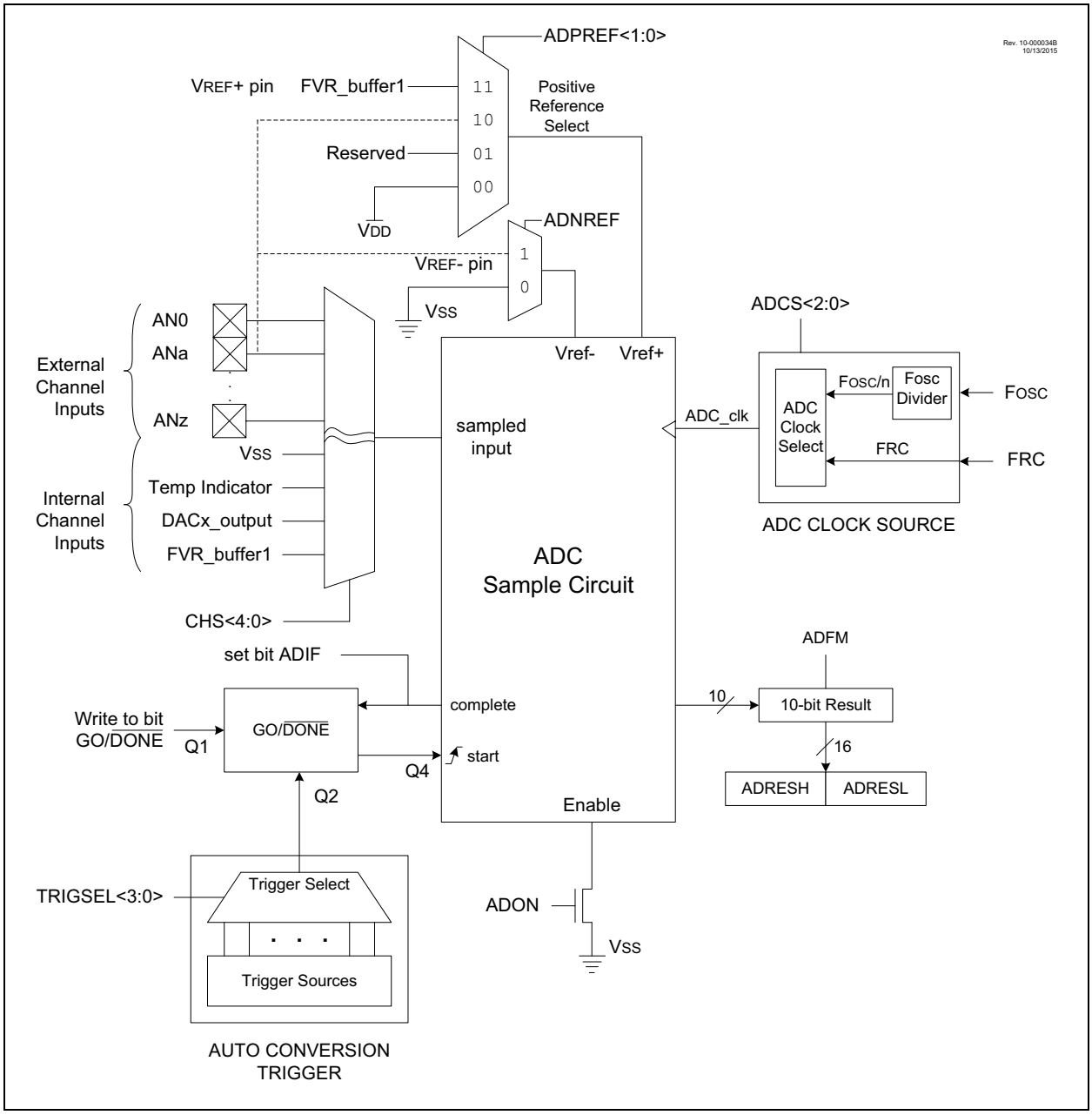


**FIGURE 23-1: ADC² BLOCK DIAGRAM**



## 25.4 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.

## 25.5 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 26-4: MDCARH: MODULATION HIGH CARRIER CONTROL REGISTER**

| U-0 | U-0 | U-0 | U-0 | R/W-x/u | R/W-x/u | R/W-x/u | R/W-x/u |
|---|---|---|---|---|---|---|---|
| — | — | — | — | MDCHS<3:0>(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 | |

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

bit 3-0 **MDCHS<3:0>** Modulator Data High Carrier Selection bits (1)

1111 = LC4_out
1110 = LC3_out
1101 = LC2_out
1100 = LC1_out
1011 = NCO output
1010 = PWM7_out
1001 = PWM6_out
1000 = CCP5 output (PWM Output mode only)
0111 = CCP4 output (PWM Output mode only)
0110 = CCP3 output (PWM Output mode only)
0101 = CCP2 output (PWM Output mode only)
0100 = CCP1 output (PWM Output mode only)
0011 = Reference clock module signal (CLKR)
0010 = HFINTOSC
0001 = FOSC
0000 = Pin selected by MDCARHPPS

**Note 1:** Narrowed carrier pulse widths or spurs may occur in the signal stream if the carrier is not synchronized.

**FIGURE 31-20: I²C SLAVE, 10-BIT ADDRESS, RECEPTION (SEN = 1, AHEN = 0, DHEN = 0)**

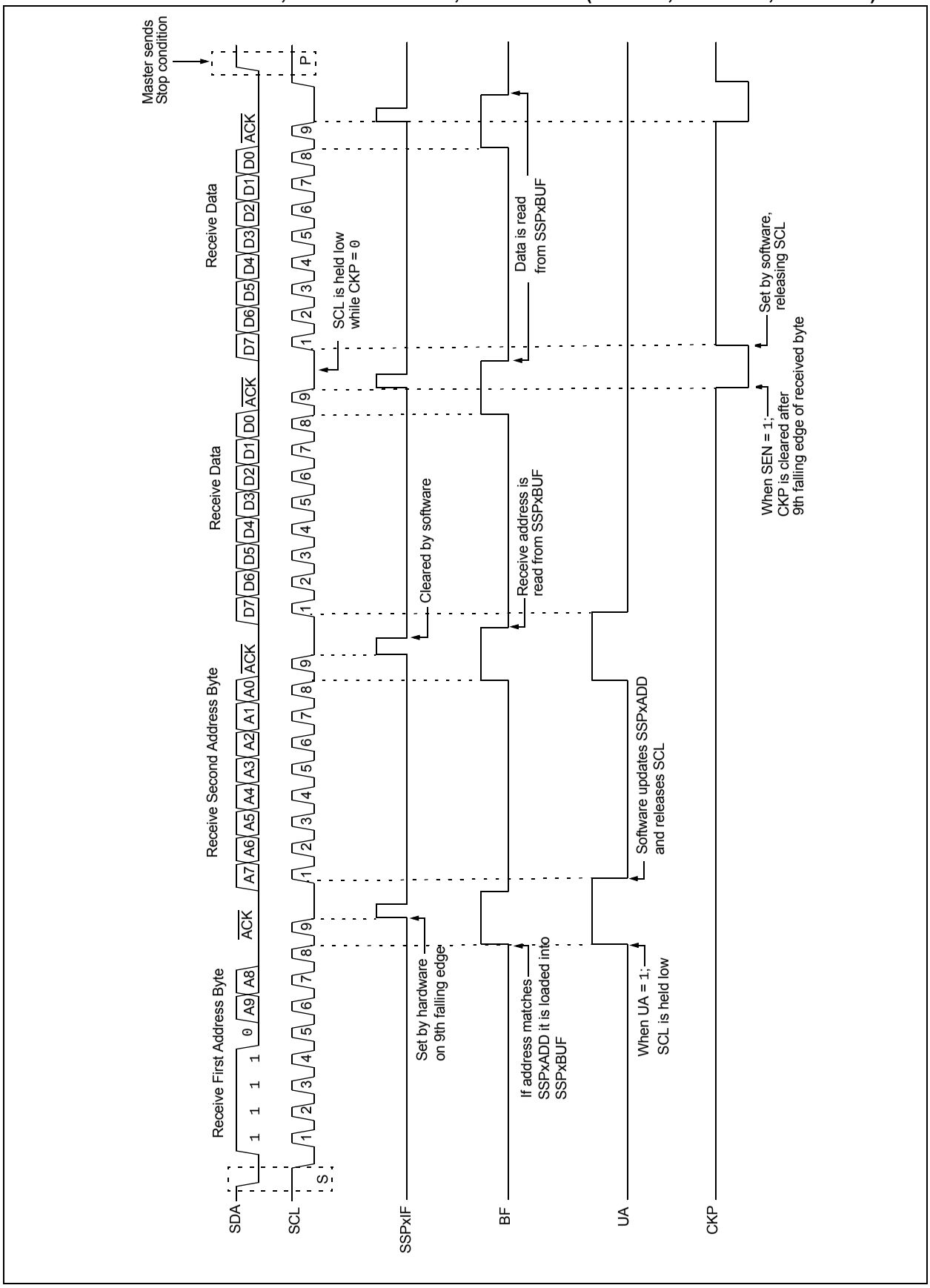
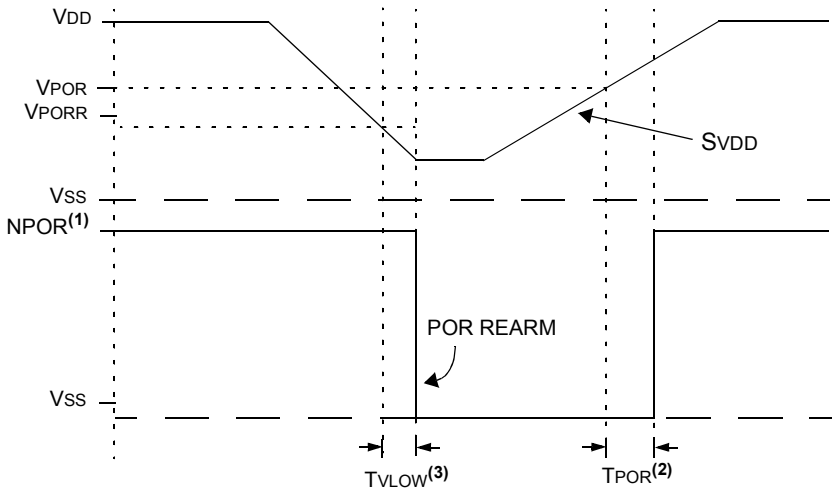


**TABLE 32-3: SUMMARY OF REGISTERS ASSOCIATED WITH SMTx**

| Name | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | Register on Page |
|---|---|---|---|---|---|---|---|---|---|
| PIE8 | — | — | SMT2PWAIE | SMT2PRAIE | SMT2IE | SMT1PWAIE | SMT1PRAIE | SMT1IE | 143 |
| PIR8 | — | — | SMT2PWAIF | SMT2PRAIF | SMT2IF | SMT1PWAIF | SMT1PRAIF | SMT1IF | 153 |
| SMT1TMRL | SMT1TMR<7:0> | | | | | | | | 549 |
| SMT1TMRH | SMT1TMR<15:8> | | | | | | | | 549 |
| SMT1TMRU | SMT1TMR<23:16> | | | | | | | | 549 |
| SMT1CPRL | SMT1CPR<7:0> | | | | | | | | 550 |
| SMT1CPRH | SMT1CPR<15:8> | | | | | | | | 550 |
| SMT1CPRU | SMT1CPR<23:16> | | | | | | | | 550 |
| SMT1CPWL | SMT1CPW<7:0> | | | | | | | | 551 |
| SMT1CPWH | SMT1CPW<15:8> | | | | | | | | 551 |
| SMT1CPWU | SMT1CPW<23:16> | | | | | | | | 551 |
| SMT1PRL | SMT1PR<7:0> | | | | | | | | 552 |
| SMT1PRH | SMT1PR<15:8> | | | | | | | | 552 |
| SMT1PRU | SMT1PR<23:16> | | | | | | | | 552 |
| SMT1CON0 | EN | — | STP | WPOL | SPOL | CPOL | SMT1PS<1:0> | | 543 |
| SMT1CON1 | SMT1GO | REPEAT | — | — | MODE<3:0> | | | | 544 |
| SMT1STAT | CPRUP | CPWUP | RST | — | — | TS | WS | AS | 545 |
| SMT1CLK | — | — | — | — | — | CSEL<2:0> | | | 546 |
| SMT1SIG | — | — | — | SSEL<4:0> | | | | | 548 |
| SMT1WIN | — | — | — | WSEL<4:0> | | | | | 547 |
| SMT2TMRL | SMT2TMR<7:0> | | | | | | | | 549 |
| SMT2TMRH | SMT2TMR<15:8> | | | | | | | | 549 |
| SMT2TMRU | SMT2TMR<23:16> | | | | | | | | 549 |
| SMT2CPRL | SMT2CPR<7:0> | | | | | | | | 550 |
| SMT2CPRH | SMT2CPR<15:8> | | | | | | | | 550 |
| SMT2CPRU | SMT2CPR<23:16> | | | | | | | | 550 |
| SMT2CPWL | SMT2CPW<7:0> | | | | | | | | 551 |
| SMT2CPWH | SMT2CPW<15:8> | | | | | | | | 551 |
| SMT2CPWU | SMT2CPW<23:16> | | | | | | | | 551 |
| SMT2PRL | SMT2PR<7:0> | | | | | | | | 552 |
| SMT2PRH | SMT2PR<15:8> | | | | | | | | 552 |
| SMT2PRU | SMT2PR<23:16> | | | | | | | | 552 |
| SMT2CON0 | EN | — | STP | WPOL | SPOL | CPOL | SMT2PS<1:0> | | 543 |
| SMT2CON1 | SMT2GO | REPEAT | — | — | MODE<3:0> | | | | 544 |
| SMT2STAT | CPRUP | CPWUP | RST | — | — | TS | WS | AS | 545 |
| SMT2CLK | — | — | — | — | — | CSEL<2:0> | | | 546 |
| SMT2SIG | — | — | — | SSEL<4:0> | | | | | 548 |
| SMT2WIN | — | — | — | WSEL<4:0> | | | | | 547 |

**Legend:** – = unimplemented read as '0'. Shaded cells are not used for SMTx module.

**FIGURE 37-3: POR AND POR REARM WITH SLOW RISING VDD**




**Note 1:** When NPOR is low, the device is held in Reset.
**2:** TPOR 1 μs typical.
**3:** TVLOW 2.7 μs typical.