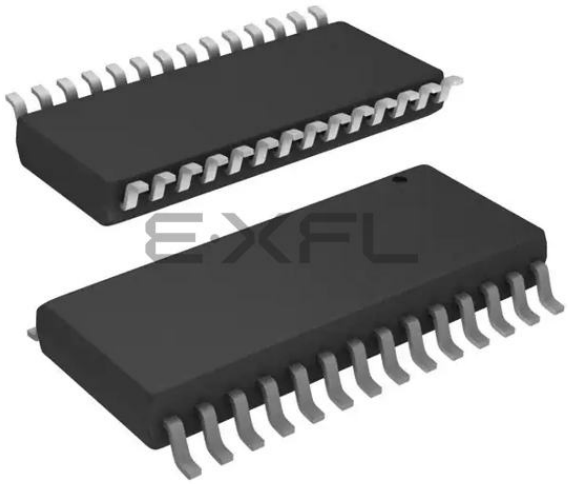


**Welcome to E-XFL.COM**

## 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 | 40MHz |
| Connectivity | CANbus, I²C, SPI, UART/USART |
| Peripherals | Brown-out Detect/Reset, HLVD, POR, PWM, WDT |
| Number of I/O | 25 |
| Program Memory Size | 48KB (24K x 16) |
| Program Memory Type | FLASH |
| EEPROM Size | 1K x 8 |
| RAM Size | 3.25K x 8 |
| Voltage - Supply (Vcc/Vdd) | 2V ~ 5.5V |
| Data Converters | A/D 8x10b |
| Oscillator Type | Internal |
| Operating Temperature | -40°C ~ 85°C (TA) |
| Mounting Type | Surface Mount |
| Package / Case | 28-SOIC (0.295", 7.50mm Width) |
| Supplier Device Package | 28-SOIC |
| Purchase URL | https://www.e-xfl.com/product-detail/microchip-technology/pic18lf2585t-i-so |

## 1.2 Other Special Features

- **Memory Endurance:** The Enhanced Flash cells for both program memory and data EEPROM are rated to last for many thousands of erase/write cycles – up to 100,000 for program memory and 1,000,000 for EEPROM. Data retention without refresh is conservatively estimated to be greater than 40 years.
- **Self-programmability:** These devices can write to their own program memory spaces under internal software control. By using a bootloader routine located in the protected Boot Block at the top of program memory, it becomes possible to create an application that can update itself in the field.
- **Extended Instruction Set:** The PIC18F2585/2680/4585/4680 family introduces an optional extension to the PIC18 instruction set, which adds 8 new instructions and an Indexed Addressing mode. This extension, enabled as a device configuration option, has been specifically designed to optimize re-entrant application code originally developed in high-level languages, such as C.
- **Enhanced CCP1 module:** In PWM mode, this module provides 1, 2 or 4 modulated outputs for controlling half-bridge and full-bridge drivers. Other features include Auto-Shutdown, for disabling PWM outputs on interrupt or other select conditions and Auto-Restart, to reactivate outputs once the condition has cleared.
- **Enhanced Addressable USART:** This serial communication module is capable of standard RS-232 operation and provides support for the LIN bus protocol. Other enhancements include automatic baud rate detection and a 16-bit Baud Rate Generator for improved resolution. When the microcontroller is using the internal oscillator block, the EUSART provides stable operation for applications that talk to the outside world without using an external crystal (or its accompanying power requirement).
- **10-bit A/D Converter:** This module incorporates programmable acquisition time, allowing for a channel to be selected and a conversion to be initiated without waiting for a sampling period and thus, reduce code overhead.
- **Extended Watchdog Timer (WDT):** This enhanced version incorporates a 16-bit prescaler, allowing a time-out range from 4 ms to over 131 seconds, that is stable across operating voltage and temperature.

## 1.3 Details on Individual Family Members

Devices in the PIC18F2585/2680/4585/4680 family are available in 28-pin (PIC18F2X8X) and 40/44-pin (PIC18F4X8X) packages. Block diagrams for the two groups are shown in Figure 1-1 and Figure 1-2.

The devices are differentiated from each other in six ways:

1. Flash program memory (48 Kbytes for PIC18FX585 devices, 64 Kbytes for PIC18FX680).
2. A/D channels (8 for PIC18F2X8X devices, 11 for PIC18F4X8X devices).
3. I/O ports (3 bidirectional ports and 1 input only port on PIC18F2X8X devices, 5 bidirectional ports on PIC18F4X8X devices).
4. CCP1 and Enhanced CCP1 implementation (PIC18F2X8X devices have 1 standard CCP1 module, PIC18F4X8X devices have one standard CCP1 module and one ECCP1 module).
5. Parallel Slave Port (present only on PIC18F4X8X devices).
6. PIC18F4X8X devices provide two comparators.

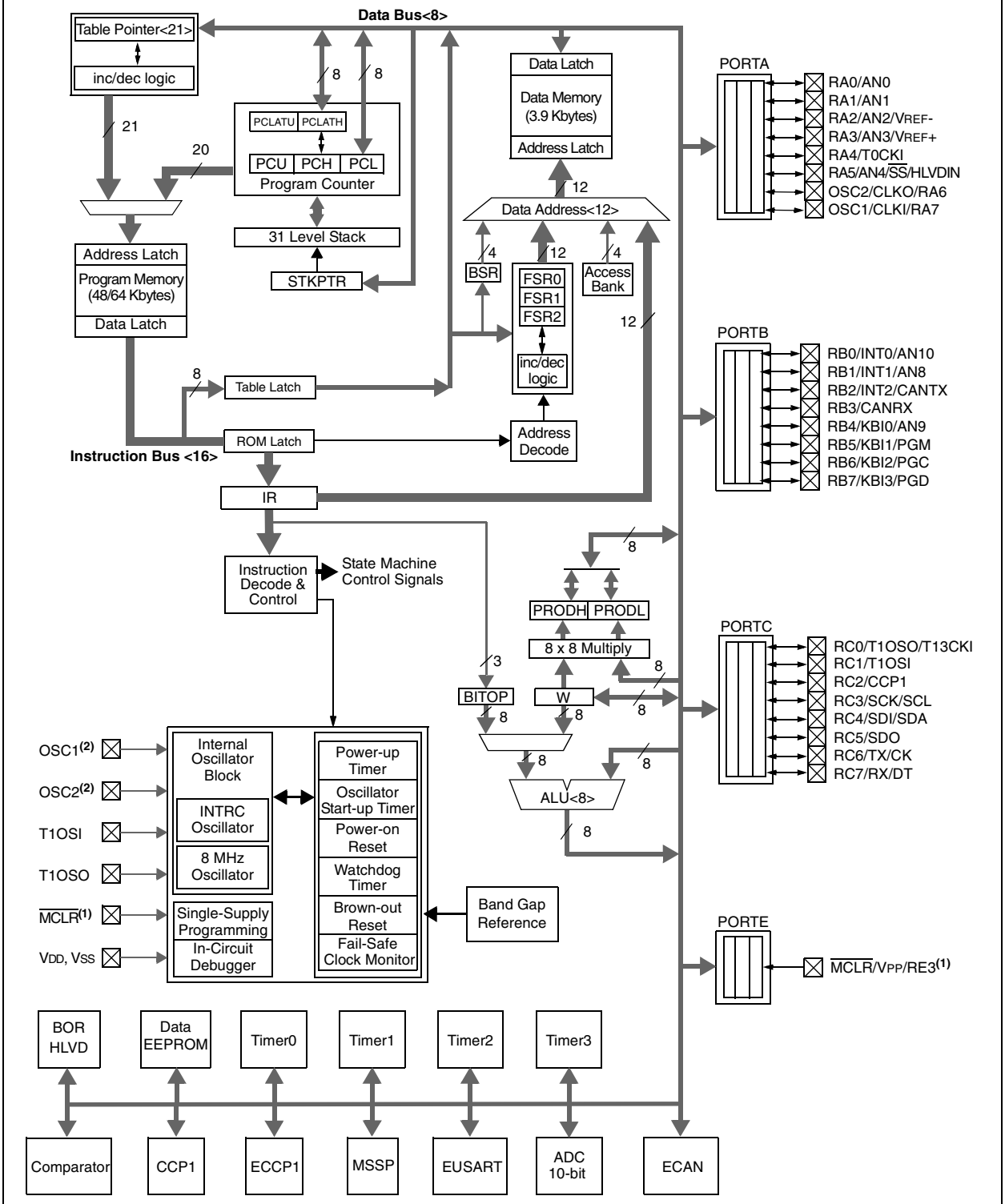
All other features for devices in this family are identical. These are summarized in Table 1-1.

The pinouts for all devices are listed in Table 1-2 and Table 1-3.

Like all Microchip PIC18 devices, members of the PIC18F2585/2680/4585/4680 family are available as both standard and low-voltage devices. Standard devices with Enhanced Flash memory, designated with an “F” in the part number (such as PIC18**F**2585), accommodate an operating $V_{DD}$ range of 4.2V to 5.5V. Low-voltage parts, designated by “LF” (such as PIC18**LF**2585), function over an extended $V_{DD}$ range of 2.0V to 5.5V.

**FIGURE 1-1: PIC18F2585/2680 (28-PIN) BLOCK DIAGRAM**




**Note 1:** RE3 is multiplexed with $\overline{\text{MCLR}}$ and is only available when the $\overline{\text{MCLR}}$ Resets are disabled.

**2:** OSC1/CLKI and OSC2/CLKO are only available in select oscillator modes and when these pins are not being used as digital I/O. Refer to **Section 2.0 "Oscillator Configurations"** for additional information.

**REGISTER 2-1: OSCTUNE: OSCILLATOR TUNING REGISTER**

| R/W-0 | R/W-0(1) | U-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 |
|---|---|---|---|---|---|---|---|
| INTSRC | PLLEN(1) | — | TUN4 | TUN3 | TUN2 | TUN1 | TUN0 |
| bit 7 | | | | | | | bit 0 |

bit 7 **INTSRC:** Internal Oscillator Low-Frequency Source Select bit
1 = 31.25 kHz device clock derived from 8 MHz INTOSC source (divide-by-256 enabled)
0 = 31 kHz device clock derived directly from INTRC internal oscillator

bit 6 **PLLEN:** Frequency Multiplier PLL for INTOSC Enable bit(1)
1 = PLL enabled for INTOSC (4 MHz and 8 MHz only)
0 = PLL disabled

**Note 1:** Available only in certain oscillator configurations; otherwise, this bit is unavailable and reads as '0'. See text for details.

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

bit 4-0 **TUN4:TUN0:** Frequency Tuning bits
01111 = Maximum frequency
•
•
00001
00000 = Center frequency. Oscillator module is running at the calibrated frequency.
11111
•
•
10000 = Minimum frequency

| **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 |

#### 2.6.5.1 Compensating with the EUSART

An adjustment may be required when the EUSART begins to generate framing errors or receives data with errors while in Asynchronous mode. Framing errors indicate that the device clock frequency is too high. To adjust for this, decrement the value in OSCTUNE to reduce the clock frequency. On the other hand, errors in data may suggest that the clock speed is too low. To compensate, increment OSCTUNE to increase the clock frequency.

#### 2.6.5.2 Compensating with the Timers

This technique compares device clock speed to some reference clock. Two timers may be used; one timer is clocked by the peripheral clock, while the other is clocked by a fixed reference source, such as the Timer1 oscillator.

Both timers are cleared, but the timer clocked by the reference generates interrupts. When an interrupt occurs, the internally clocked timer is read and both timers are cleared. If the internally clocked timer value is greater than expected, then the internal oscillator block is running too fast. To adjust for this, decrement the OSCTUNE register.

#### 2.6.5.3 Compensating with the CCP1 Module in Capture Mode

The CCP1 module can use free running Timer1 (or Timer3), clocked by the internal oscillator block and an external event with a known period (i.e., AC power frequency). The time of the first event is captured in the CCPRxH:CCPRxL registers and is recorded for use later. When the second event causes a capture, the time of the first event is subtracted from the time of the second event. Since the period of the external event is known, the time difference between events can be calculated.

If the measured time is much greater than the calculated time, the internal oscillator block is running too fast. To compensate, decrement the OSCTUNE register. If the measured time is much less than the calculated time, the internal oscillator block is running too slow. To compensate, increment the OSCTUNE register.

**REGISTER 2-2: OSCCON: OSCILLATOR CONTROL REGISTER**

| R/W-0 | R/W-1 | R/W-0 | R/W-0 | R[(1)] | R-0 | R/W-0 | R/W-0 |
|---|---|---|---|---|---|---|---|
| IDLEN | IRCF2 | IRCF1 | IRCF0 | OSTS | IOFS | SCS1 | SCS0 |
| bit 7 | | | | | | | bit 0 |

bit 7 **IDLEN:** Idle Enable bit
1 = Device enters Idle mode on SLEEP instruction
0 = Device enters Sleep mode on SLEEP instruction

bit 6-4 **IRCF2:IRCF0:** Internal Oscillator Frequency Select bits
111 = 8 MHz (INTOSC drives clock directly)
110 = 4 MHz
101 = 2 MHz
100 = 1 MHz[(3)]
011 = 500 kHz
010 = 250 kHz
001 = 125 kHz
000 = 31 kHz (from either INTOSC/256 or INTRC directly)[(2)]

bit 3 **OSTS:** Oscillator Start-up Time-out Status bit[(1)]
1 = Oscillator start-up time-out timer has expired; primary oscillator is running
0 = Oscillator start-up time-out timer is running; primary oscillator is not ready

bit 2 **IOFS:** INTOSC Frequency Stable bit
1 = INTOSC frequency is stable and the frequency is provided by one of the RC modes
0 = INTOSC frequency is not stable

bit 1-0 **SCS1:SCS0:** System Clock Select bits
1x = Internal oscillator block
01 = Timer1 oscillator
00 = Primary oscillator

**Note 1:** Depends on state of the IESO Configuration bit.
**2:** Source selected by the INTSRC bit (OSCTUNE<7>), see text.
**3:** Default output frequency of INTOSC on Reset.

| **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 |

**TABLE 4-4: INITIALIZATION CONDITIONS FOR ALL REGISTERS (CONTINUED)**

| Register | Applicable Devices | | | | Power-on Reset, Brown-out Reset | MCLR Resets, WDT Reset, RESET Instruction, Stack Resets | Wake-up via WDT or Interrupt |
|---|---|---|---|---|---|---|---|
| B4EIDH(6) | 2585 | 2680 | 4585 | 4680 | xxxx xxxx | uuuu uuuu | uuuu uuuu |
| B4SIDL(6) | 2585 | 2680 | 4585 | 4680 | xxxx x-xx | uuuu u-uu | uuuu u-uu |
| B4SIDH(6) | 2585 | 2680 | 4585 | 4680 | xxxx xxxx | uuuu uuuu | uuuu uuuu |
| B4CON(6) | 2585 | 2680 | 4585 | 4680 | 0000 0000 | 0000 0000 | uuuu uuuu |
| B3D7(6) | 2585 | 2680 | 4585 | 4680 | xxxx xxxx | uuuu uuuu | uuuu uuuu |
| B3D6(6) | 2585 | 2680 | 4585 | 4680 | xxxx xxxx | uuuu uuuu | uuuu uuuu |
| B3D5(6) | 2585 | 2680 | 4585 | 4680 | xxxx xxxx | uuuu uuuu | uuuu uuuu |
| B3D4(6) | 2585 | 2680 | 4585 | 4680 | xxxx xxxx | uuuu uuuu | uuuu uuuu |
| B3D3(6) | 2585 | 2680 | 4585 | 4680 | xxxx xxxx | uuuu uuuu | uuuu uuuu |
| B3D2(6) | 2585 | 2680 | 4585 | 4680 | xxxx xxxx | uuuu uuuu | uuuu uuuu |
| B3D1(6) | 2585 | 2680 | 4585 | 4680 | xxxx xxxx | uuuu uuuu | uuuu uuuu |
| B3D0(6) | 2585 | 2680 | 4585 | 4680 | xxxx xxxx | uuuu uuuu | uuuu uuuu |
| B3DLC(6) | 2585 | 2680 | 4585 | 4680 | -xxx xxxx | -uuu uuuu | -uuu uuuu |
| B3EIDL(6) | 2585 | 2680 | 4585 | 4680 | xxxx xxxx | uuuu uuuu | uuuu uuuu |
| B3EIDH(6) | 2585 | 2680 | 4585 | 4680 | xxxx xxxx | uuuu uuuu | uuuu uuuu |
| B3SIDL(6) | 2585 | 2680 | 4585 | 4680 | xxxx x-xx | uuuu u-uu | uuuu u-uu |
| B3SIDH(6) | 2585 | 2680 | 4585 | 4680 | xxxx xxxx | uuuu uuuu | uuuu uuuu |
| B3CON(6) | 2585 | 2680 | 4585 | 4680 | 0000 0000 | 0000 0000 | uuuu uuuu |
| B2D7(6) | 2585 | 2680 | 4585 | 4680 | xxxx xxxx | uuuu uuuu | uuuu uuuu |
| B2D6(6) | 2585 | 2680 | 4585 | 4680 | xxxx xxxx | uuuu uuuu | uuuu uuuu |
| B2D5(6) | 2585 | 2680 | 4585 | 4680 | xxxx xxxx | uuuu uuuu | uuuu uuuu |
| B2D4(6) | 2585 | 2680 | 4585 | 4680 | xxxx xxxx | uuuu uuuu | uuuu uuuu |
| B2D3(6) | 2585 | 2680 | 4585 | 4680 | xxxx xxxx | uuuu uuuu | uuuu uuuu |
| B2D2(6) | 2585 | 2680 | 4585 | 4680 | xxxx xxxx | uuuu uuuu | uuuu uuuu |
| B2D1(6) | 2585 | 2680 | 4585 | 4680 | xxxx xxxx | uuuu uuuu | uuuu uuuu |
| B2D0(6) | 2585 | 2680 | 4585 | 4680 | xxxx xxxx | uuuu uuuu | uuuu uuuu |
| B2DLC(6) | 2585 | 2680 | 4585 | 4680 | -xxx xxxx | -uuu uuuu | -uuu uuuu |
| B2EIDL(6) | 2585 | 2680 | 4585 | 4680 | xxxx xxxx | uuuu uuuu | uuuu uuuu |
| B2EIDH(6) | 2585 | 2680 | 4585 | 4680 | xxxx xxxx | uuuu uuuu | uuuu uuuu |
| B2SIDL(6) | 2585 | 2680 | 4585 | 4680 | xxxx x-xx | uuuu u-uu | uuuu u-uu |
| B2SIDH(6) | 2585 | 2680 | 4585 | 4680 | xxxx xxxx | uuuu uuuu | uuuu uuuu |

**Legend:** u = unchanged, x = unknown, - = unimplemented bit, read as '0', q = value depends on condition. Shaded cells indicate conditions do not apply for the designated device.

**Note 1:** One or more bits in the INTCONx or PIRx registers will be affected (to cause wake-up).

**2:** When the wake-up is due to an interrupt and the GIEL or GIEH bit is set, the PC is loaded with the interrupt vector (0008h or 0018h).

**3:** When the wake-up is due to an interrupt and the GIEL or GIEH bit is set, the TOSU, TOSH and TOSL are updated with the current value of the PC. The STKPTR is modified to point to the next location in the hardware stack.

**4:** See Table 4-3 for Reset value for specific condition.

**5:** Bits 6 and 7 of PORTA, LATA and TRISA are enabled, depending on the oscillator mode selected. When not enabled as PORTA pins, they are disabled and read '0'.

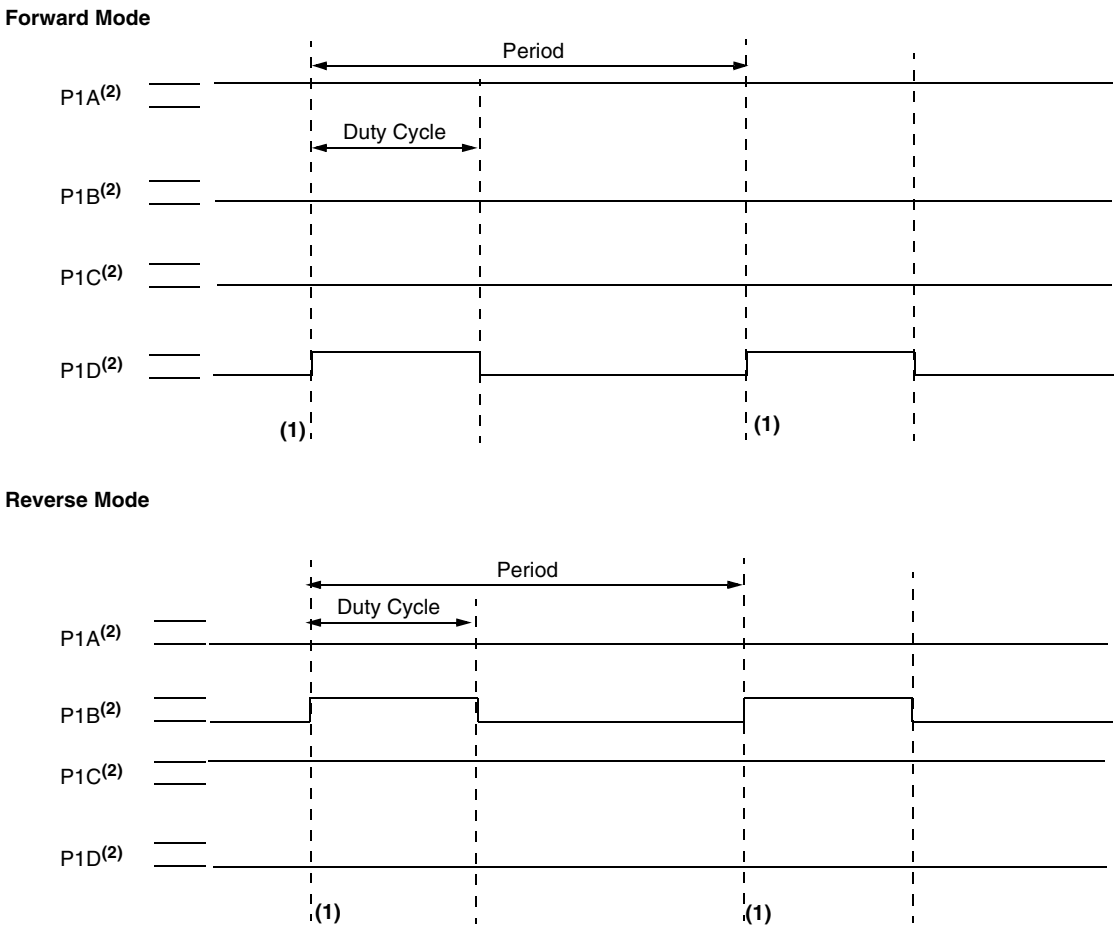
**6:** This register reads all '0's until the ECAN™ technology is set up in Mode 1 or Mode 2.

### 16.4.5 FULL-BRIDGE MODE

In Full-Bridge Output mode, four pins are used as outputs; however, only two outputs are active at a time. In the Forward mode, pin P1A is continuously active and pin P1D is modulated. In the Reverse mode, pin P1C is continuously active and pin P1B is modulated. These are illustrated in Figure 16-6.

P1A, P1B, P1C and P1D outputs are multiplexed with the PORTD<4>, PORTD<5>, PORTD<6> and PORTD<7> data latches. The TRISD<4>, TRISD<5>, TRISD<6> and TRISD<7> bits must be cleared to make the P1A, P1B, P1C and P1D pins outputs.

**FIGURE 16-6: FULL-BRIDGE PWM OUTPUT**




**Note 1:** At this time, the TMR2 register is equal to the PR2 register.
**2:** Output signal is shown as active-high.

## 20.0 COMPARATOR MODULE

The analog comparator module contains two comparators that can be configured in a variety of ways. The inputs can be selected from the analog inputs multiplexed with pins RA0 through RA5, as well as the on-chip voltage reference (see **Section 21.0 "Comparator Voltage Reference Module"**). The digital outputs (normal or inverted) are available at the pin level and can also be read through the control register.

The CMCON register (Register 20-1) selects the comparator input and output configuration. Block diagrams of the various comparator configurations are shown in Figure 20-1.

**REGISTER 20-1: CMCON: COMPARATOR CONTROL REGISTER**

| R-0 | R-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 |
|---|---|---|---|---|---|---|---|
| C2OUT | C1OUT | C2INV | C1INV | CIS | CM2 | CM1 | CM0 |
| bit 7 | | | | | | | bit 0 |

bit 7 **C2OUT**: Comparator 2 Output bit
When C2INV = 0:
1 = C2 VIN+ > C2 VIN-
0 = C2 VIN+ < C2 VIN-
When C2INV = 1:
1 = C2 VIN+ < C2 VIN-
0 = C2 VIN+ > C2 VIN-

bit 6 **C1OUT**: Comparator 1 Output bit
When C1INV = 0:
1 = C1 VIN+ > C1 VIN-
0 = C1 VIN+ < C1 VIN-
When C1INV = 1:
1 = C1 VIN+ < C1 VIN-
0 = C1 VIN+ > C1 VIN-

bit 5 **C2INV**: Comparator 2 Output Inversion bit
1 = C2 output inverted
0 = C2 output not inverted

bit 4 **C1INV**: Comparator 1 Output Inversion bit
1 = C1 Output inverted
0 = C1 Output not inverted

bit 3 **CIS**: Comparator Input Switch bit
When CM2:CM0 = 110:
1 = C1 VIN- connects to RD0/PSP0/C1IN+
C2 VIN- connects to RD2/PSP2/C2IN+
0 = C1 VIN- connects to RD1/PSP1/C1IN-
C2 VIN- connects to RD3/PSP3/C2IN-

bit 2-0 **CM2:CM0**: Comparator Mode bits
Figure 20-1 shows the Comparator modes and the CM2:CM0 bit settings.

| **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 |

### 23.2.2 DEDICATED CAN TRANSMIT BUFFER REGISTERS

This section describes the dedicated CAN Transmit Buffer registers and their associated control registers.

**REGISTER 23-5: TXBnCON: TRANSMIT BUFFER n CONTROL REGISTERS [0 ≤ n ≤ 2]**

**Mode 0**

| R/C-0 | R-0 | R-0 | R-0 | R/W-0 | U-0 | R/W-0 | R/W-0 |
|---|---|---|---|---|---|---|---|
| TXBIF | TXABT(1) | TXLARB(1) | TXERR(1) | TXREQ(2) | — | TXPRI1(3) | TXPRI0(3) |

**Mode 1, 2**

| R/C-0 | R-0 | R-0 | R-0 | R/W-0 | U-0 | R/W-0 | R/W-0 |
|---|---|---|---|---|---|---|---|
| TXBIF | TXABT(1) | TXLARB(1) | TXERR(1) | TXREQ(2) | — | TXPRI1(3) | TXPRI0(3) |
| bit 7 | | | | | | | bit 0 |

bit 7 **TXBIF:** Transmit Buffer Interrupt Flag bit
1 = Transmit buffer has completed transmission of message and may be reloaded
0 = Transmit buffer has not completed transmission of a message

bit 6 **TXABT:** Transmission Aborted Status bit(1)
1 = Message was aborted
0 = Message was not aborted

bit 5 **TXLARB:** Transmission Lost Arbitration Status bit(1)
1 = Message lost arbitration while being sent
0 = Message did not lose arbitration while being sent

bit 4 **TXERR:** Transmission Error Detected Status bit(1)
1 = A bus error occurred while the message was being sent
0 = A bus error did not occur while the message was being sent

bit 3 **TXREQ:** Transmit Request Status bit(2)
1 = Requests sending a message. Clears the TXABT, TXLARB and TXERR bits.
0 = Automatically cleared when the message is successfully sent

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

bit 1-0 **TXPRI1:TXPRI0:** Transmit Priority bits(3)
11 = Priority Level 3 (highest priority)
10 = Priority Level 2
01 = Priority Level 1
00 = Priority Level 0 (lowest priority)

**Note 1:** This bit is automatically cleared when TXREQ is set.

**2:** While TXREQ is set, Transmit Buffer registers remain read-only. Clearing this bit in software while the bit is set will request a message abort.

**3:** These bits define the order in which transmit buffers will be transferred. They do not alter the CAN message identifier.

| **Legend:** | | | |
|---|---|---|---|
| C = Clearable bit | 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 |

**REGISTER 23-24: BnSIDH: TX/RX BUFFER n STANDARD IDENTIFIER REGISTERS, HIGH BYTE IN RECEIVE MODE [0 ≤ n ≤ 5, TXnEN (BSEL0<n>) = 0](1)**

| R-x | R-x | R-x | R-x | R-x | R-x | R-x | R-x |
|---|---|---|---|---|---|---|---|
| SID10 | SID9 | SID8 | SID7 | SID6 | SID5 | SID4 | SID3 |
| bit 7 | | | | | | | bit 0 |

bit 7-0 **SID10:SID3:** Standard Identifier bits (if EXIDE (BnSIDL<3>) = 0)
Extended Identifier bits EID28:EID21 (if EXIDE = 1).

**Note 1:** These registers are available in Mode 1 and 2 only.

| **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 |

**REGISTER 23-25: BnSIDH: TX/RX BUFFER n STANDARD IDENTIFIER REGISTERS, HIGH BYTE IN TRANSMIT MODE [0 ≤ n ≤ 5, TXnEN (BSEL0<n>) = 1](1)**

| R/W-x | R/W-x | R/W-x | R/W-x | R/W-x | R/W-x | R/W-x | R/W-x |
|---|---|---|---|---|---|---|---|
| SID10 | SID9 | SID8 | SID7 | SID6 | SID5 | SID4 | SID3 |
| bit 7 | | | | | | | bit 0 |

bit 7-0 **SID10:SID3:** Standard Identifier bits (if EXIDE (BnSIDL<3>) = 0)
Extended Identifier bits EID28:EID21 (if EXIDE = 1).

**Note 1:** These registers are available in Mode 1 and 2 only.

| **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 |

**REGISTER 23-50: MSEL2: MASK SELECT REGISTER 2(1)**

| R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 |
|---|---|---|---|---|---|---|---|
| FIL11_1 | FIL11_0 | FIL10_1 | FIL10_0 | FIL9_1 | FIL9_0 | FIL8_1 | FIL8_0 |
| bit 7 | | | | | | | bit 0 |

bit 7-6 **FIL11_1:FIL11_0:** Filter 11 Select bits 1 and 0
11 = No mask
10 = Filter 15
01 = Acceptance Mask 1
00 = Acceptance Mask 0

bit 5-4 **FIL10_1:FIL10_0:** Filter 10 Select bits 1 and 0
11 = No mask
10 = Filter 15
01 = Acceptance Mask 1
00 = Acceptance Mask 0

bit 3-2 **FIL9_1:FIL9_0:** Filter 9 Select bits 1 and 0
11 = No mask
10 = Filter 15
01 = Acceptance Mask 1
00 = Acceptance Mask 0

bit 1-0 **FIL8_1:FIL8_0:** Filter 8 Select bits 1 and 0
11 = No mask
10 = Filter 15
01 = Acceptance Mask 1
00 = Acceptance Mask 0

**Note 1:** This register is available in Mode 1 and 2 only.

| **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 |

**TABLE 23-1: CAN CONTROLLER REGISTER MAP (CONTINUED)**

| Address[1] | Name | Address | Name | Address | Name | Address | Name |
|---|---|---|---|---|---|---|---|
| DFFh | —[4] | DDFh | —[4] | DBFh | —[4] | D9Fh | —[4] |
| DFEh | —[4] | DDEh | —[4] | DBEh | —[4] | D9Eh | —[4] |
| DFDh | —[4] | DDDh | —[4] | DBDh | —[4] | D9Dh | —[4] |
| DFCh | TXBIE | DDCh | —[4] | DBCh | —[4] | D9Ch | —[4] |
| DFBh | —[4] | DDBh | —[4] | DBBh | —[4] | D9Bh | —[4] |
| DFAh | BIE0 | DDAh | —[4] | DBAh | —[4] | D9Ah | —[4] |
| DF9h | —[4] | DD9h | —[4] | DB9h | —[4] | D99h | —[4] |
| DF8h | BSEL0 | DD8h | SDFLC | DB8h | —[4] | D98h | —[4] |
| DF7h | —[4] | DD7h | —[4] | DB7h | —[4] | D97h | —[4] |
| DF6h | —[4] | DD6h | —[4] | DB6h | —[4] | D96h | —[4] |
| DF5h | —[4] | DD5h | RXFCON1 | DB5h | —[4] | D95h | —[4] |
| DF4h | —[4] | DD4h | RXFCON0 | DB4h | —[4] | D94h | —[4] |
| DF3h | MSEL3 | DD3h | —[4] | DB3h | —[4] | D93h | RXF15EIDL |
| DF2h | MSEL2 | DD2h | —[4] | DB2h | —[4] | D92h | RXF15EIDH |
| DF1h | MSEL1 | DD1h | —[4] | DB1h | —[4] | D91h | RXF15SIDL |
| DF0h | MSEL0 | DD0h | —[4] | DB0h | —[4] | D90h | RXF15SIDH |
| DEFh | —[4] | DCFh | —[4] | DAFh | —[4] | D8Fh | —[4] |
| DEEh | —[4] | DCEh | —[4] | DAEh | —[4] | D8Eh | —[4] |
| DEDh | —[4] | DCDh | —[4] | DADh | —[4] | D8Dh | —[4] |
| DECh | —[4] | DCCh | —[4] | DACh | —[4] | D8Ch | —[4] |
| DEBh | —[4] | DCBh | —[4] | DABh | —[4] | D8Bh | RXF14EIDL |
| DEAh | —[4] | DCAh | —[4] | DAAh | —[4] | D8Ah | RXF14EIDH |
| DE9h | —[4] | DC9h | —[4] | DA9h | —[4] | D89h | RXF14SIDL |
| DE8h | —[4] | DC8h | —[4] | DA8h | —[4] | D88h | RXF14SIDH |
| DE7h | RXFBCON7 | DC7h | —[4] | DA7h | —[4] | D87h | RXF13EIDL |
| DE6h | RXFBCON6 | DC6h | —[4] | DA6h | —[4] | D86h | RXF13EIDH |
| DE5h | RXFBCON5 | DC5h | —[4] | DA5h | —[4] | D85h | RXF13SIDL |
| DE4h | RXFBCON4 | DC4h | —[4] | DA4h | —[4] | D84h | RXF13SIDH |
| DE3h | RXFBCON3 | DC3h | —[4] | DA3h | —[4] | D83h | RXF12EIDL |
| DE2h | RXFBCON2 | DC2h | —[4] | DA2h | —[4] | D82h | RXF12EIDH |
| DE1h | RXFBCON1 | DC1h | —[4] | DA1h | —[4] | D81h | RXF12SIDL |
| DE0h | RXFBCON0 | DC0h | —[4] | DA0h | —[4] | D80h | RXF12SIDH |

**Note 1:** Shaded registers are available in Access Bank low area, while the rest are available in Bank 15.

**2:** CANSTAT register is repeated in these locations to simplify application firmware. Unique names are given for each instance of the controller register due to the Microchip header file requirement.

**3:** These registers are not CAN registers.

**4:** Unimplemented registers are read as '0'.

### 23.3.4 LISTEN ONLY MODE

Listen Only mode provides a means for the PIC18F2585/2680/4585/4680 devices to receive all messages, including messages with errors. This mode can be used for bus monitor applications or for detecting the baud rate in 'hot plugging' situations. For Auto-Baud Detection, it is necessary that there are at least two other nodes which are communicating with each other. The baud rate can be detected empirically by testing different values until valid messages are received. The Listen Only mode is a silent mode, meaning no messages will be transmitted while in this state, including error flags or Acknowledge signals. The filters and masks can be used to allow only particular messages to be loaded into the receive registers or the filter masks can be set to all zeros to allow a message with any identifier to pass. The error counters are reset and deactivated in this state. The Listen Only mode is activated by setting the mode request bits in the CANCON register.

### 23.3.5 LOOPBACK MODE

This mode will allow internal transmission of messages from the transmit buffers to the receive buffers without actually transmitting messages on the CAN bus. This mode can be used in system development and testing. In this mode, the ACK bit is ignored and the device will allow incoming messages from itself, just as if they were coming from another node. The Loopback mode is a silent mode, meaning no messages will be transmitted while in this state, including error flags or Acknowledge signals. The TXCAN pin will revert to port I/O while the device is in this mode. The filters and masks can be used to allow only particular messages to be loaded into the receive registers. The masks can be set to all zeros to provide a mode that accepts all messages. The Loopback mode is activated by setting the mode request bits in the CANCON register.

### 23.3.6 ERROR RECOGNITION MODE

The module can be set to ignore all errors and receive any message. In functional Mode 0, the Error Recognition mode is activated by setting the RXM<1:0> bits in the RXBnCON registers to '11'. In this mode, the data which is in the message assembly buffer until the error time, is copied in the receive buffer and can be read via the CPU interface.

## 23.4 ECAN Module Functional Modes

In addition to CAN modes of operation, the ECAN module offers a total of 3 functional modes. Each of these modes are identified as Mode 0, Mode 1 and Mode 2.

### 23.4.1 MODE 0 – LEGACY MODE

Mode 0 is designed to be fully compatible with CAN modules used in PIC18CXX8 and PIC18FXX8 devices. This is the default mode of operation on all Reset conditions. As a result, module code written for the PIC18XX8 CAN module may be used on the ECAN module without any code changes.

The following is the list of resources available in Mode 0:

- Three transmit buffers: TXB0, TXB1 and TXB2
- Two receive buffers: RXB0 and RXB1
- Two acceptance masks, one for each receive buffer: RXM0, RXM1
- Six acceptance filters, 2 for RXB0 and 4 for RXB1: RXF0, RXF1, RXF2, RXF3, RXF4, RXF5

### 23.4.2 MODE 1 – ENHANCED LEGACY MODE

Mode 1 is similar to Mode 0, with the exception that more resources are available in Mode 1. There are 16 acceptance filters and two acceptance mask registers. Acceptance Filter 15 can be used as either an acceptance filter or an acceptance mask register. In addition to three transmit and two receive buffers, there are six more message buffers. One or more of these additional buffers can be programmed as transmit or receive buffers. These additional buffers can also be programmed to automatically handle RTR messages.

Fourteen of sixteen acceptance filter registers can be dynamically associated to any receive buffer and acceptance mask register. One can use this capability to associate more than one filter to any one buffer.

When a receive buffer is programmed to use standard identifier messages, part of the full acceptance filter register can be used as a data byte filter. The length of the data byte filter is programmable from 0 to 18 bits. This functionality simplifies implementation of high-level protocols, such as the DeviceNet protocol.

The following is the list of resources available in Mode 1:

- Three transmit buffers: TXB0, TXB1 and TXB2
- Two receive buffers: RXB0 and RXB1
- Six buffers programmable as TX or RX: B0-B5
- Automatic RTR handling on B0-B5
- Sixteen dynamically assigned acceptance filters: RXF0-RXF15
- Two dedicated acceptance mask registers; RXF15 programmable as third mask: RXM0-RXM1, RXF15
- Programmable data filter on standard identifier messages: SDFLC

### 23.9.2 TIME QUANTA

As already mentioned, the Time Quanta is a fixed unit derived from the oscillator period and baud rate prescaler. Its relationship to TBIT and the Nominal Bit Rate is shown in Example 23-6.

**EXAMPLE 23-6: CALCULATING TQ, NOMINAL BIT RATE AND NOMINAL BIT TIME**

TQ (µs) = (2 * (BRP + 1))/FOSC (MHz)

TBIT (µs) = TQ (µs) * number of TQ per bit interval

Nominal Bit Rate (bits/s) = 1/TBIT

This frequency (FOSC) refers to the effective frequency used. If, for example, a 10 MHz external signal is used along with a PLL, then the effective frequency will be 4 x 10 MHz which equals 40 MHz.

**CASE 1:**

*For FOSC = 16 MHz, BRP<5:0> = 00h and Nominal Bit Time = 8 TQ:*

TQ = (2 * 1)/16 = 0.125 µs (125 ns)

TBIT = 8 * 0.125 = 1 µs ($10^{-6}$s)

Nominal Bit Rate = $1/10^{-6} = 10^6$ bits/s (1 Mb/s)

**CASE 2:**

*For FOSC = 20 MHz, BRP<5:0> = 01h and Nominal Bit Time = 8 TQ:*

TQ = (2 * 2)/20 = 0.2 µs (200 ns)

TBIT = 8 * 0.2 = 1.6 µs ($1.6 * 10^{-6}$s)

Nominal Bit Rate = $1/1.6 * 10^{-6}$s = 625,000 bits/s (625 Kb/s)

**CASE 3:**

*For FOSC = 25 MHz, BRP<5:0> = 3Fh and Nominal Bit Time = 25 TQ:*

TQ = (2 * 64)/25 = 5.12 µs

TBIT = 25 * 5.12 = 128 µs ($1.28 * 10^{-4}$s)

Nominal Bit Rate = $1/1.28 * 10^{-4}$ = 7813 bits/s (7.8 Kb/s)

The frequencies of the oscillators in the different nodes must be coordinated in order to provide a system wide specified nominal bit time. This means that all oscillators must have a TOSC that is an integral divisor of TQ. It should also be noted that although the number of TQ is programmable from 4 to 25, the usable minimum is 8 TQ. There is no assurance that a bit time of less than 8 TQ in length will operate correctly.

### 23.9.3 SYNCHRONIZATION SEGMENT

This part of the bit time is used to synchronize the various CAN nodes on the bus. The edge of the input signal is expected to occur during the sync segment. The duration is 1 TQ.

### 23.9.4 PROPAGATION SEGMENT

This part of the bit time is used to compensate for physical delay times within the network. These delay times consist of the signal propagation time on the bus line and the internal delay time of the nodes. The length of the Propagation Segment can be programmed from 1 TQ to 8 TQ by setting the PRSEG2:PRSEG0 bits.

### 23.9.5 PHASE BUFFER SEGMENTS

The phase buffer segments are used to optimally locate the sampling point of the received bit within the nominal bit time. The sampling point occurs between Phase Segment 1 and Phase Segment 2. These segments can be lengthened or shortened by the resynchronization process. The end of Phase Segment 1 determines the sampling point within a bit time. Phase Segment 1 is programmable from 1 TQ to 8 TQ in duration. Phase Segment 2 provides delay before the next transmitted data transition and is also programmable from 1 TQ to 8 TQ in duration. However, due to IPT requirements, the actual minimum length of Phase Segment 2 is 2 TQ, or it may be defined to be equal to the greater of Phase Segment 1 or the Information Processing Time (IPT). The sampling point should be as late as possible or approximately 80% of the bit time.

### 23.9.6 SAMPLE POINT

The sample point is the point of time at which the bus level is read and the value of the received bit is determined. The sampling point occurs at the end of Phase Segment 1. If the bit timing is slow and contains many TQ, it is possible to specify multiple sampling of the bus line at the sample point. The value of the received bit is determined to be the value of the majority decision of three values. The three samples are taken at the sample point and twice before, with a time of TQ/2 between each sample.

### 23.9.7 INFORMATION PROCESSING TIME

The Information Processing Time (IPT) is the time segment starting at the sample point that is reserved for calculation of the subsequent bit level. The CAN specification defines this time to be less than or equal to 2 TQ. The PIC18F2585/2680/4585/4680 devices define this time to be 2 TQ. Thus, Phase Segment 2 must be at least 2 TQ long.

**REGISTER 24-3: CONFIG2H: CONFIGURATION REGISTER 2 HIGH (BYTE ADDRESS 300003h)**

| U-0 | U-0 | U-0 | R/P-1 | R/P-1 | R/P-1 | R/P-1 | R/P-1 |
|---|---|---|---|---|---|---|---|
| — | — | — | WDTPS3 | WDTPS2 | WDTPS1 | WDTPS0 | WDTEN |
| bit 7 | | | | | | | bit 0 |

bit 7-5 **Unimplemented:** Read as '0'

bit 4-1 **WDTPS3:WDTPS0:** Watchdog Timer Postscale Select bits

1111 = 1:32,768
1110 = 1:16,384
1101 = 1:8,192
1100 = 1:4,096
1011 = 1:2,048
1010 = 1:1,024
1001 = 1:512
1000 = 1:256
0111 = 1:128
0110 = 1:64
0101 = 1:32
0100 = 1:16
0011 = 1:8
0010 = 1:4
0001 = 1:2
0000 = 1:1

bit 0 **WDTEN:** Watchdog Timer Enable bit

1 = WDT enabled
0 = WDT disabled (control is placed on the SWDTEN bit)

| **Legend:** | | |
|---|---|---|
| R = Readable bit | P = Programmable bit | U = Unimplemented bit, read as '0' |
| -n = Value when device is unprogrammed | | u = Unchanged from programmed state |

**REGISTER 24-12: DEVICE ID REGISTER 1 FOR PIC18F2585/2680/4585/4680 DEVICES**

| R | R | R | R | R | R | R | R |
|---|---|---|---|---|---|---|---|
| DEV2 | DEV1 | DEV0 | REV4 | REV3 | REV2 | REV1 | REV0 |
| bit 7 | | | | | | | bit 0 |

bit 7-5 **DEV2:DEV0:** Device ID bits

111 = PIC18F2585
110 = PIC18F2680
101 = PIC18F4585
100 = PIC18F4680

bit 4-0 **REV4:REV0:** Revision ID bits

These bits are used to indicate the device revision.

| **Legend:** | | |
|---|---|---|
| R = Read-only bit | P = Programmable bit | U = Unimplemented bit, read as '0' |
| -n = Value when device is unprogrammed | | u = Unchanged from programmed state |

**REGISTER 24-13: DEVICE ID REGISTER 2 FOR PIC18F2585/2680/4585/4680 DEVICES**

| R | R | R | R | R | R | R | R |
|---|---|---|---|---|---|---|---|
| DEV10 | DEV9 | DEV8 | DEV7 | DEV6 | DEV5 | DEV4 | DEV3 |
| bit 7 | | | | | | | bit 0 |

bit 7-0 **DEV10:DEV3:** Device ID bits

These bits are used with the DEV2:DEV0 bits in the Device ID Register 1 to identify the part number.

0000 1110 = PIC18F2585/2680/4585/4680 devices

**Note:** These values for DEV10:DEV3 may be shared with other devices. The specific device is always identified by using the entire DEV10:DEV0 bit sequence.

| **Legend:** | | |
|---|---|---|
| R = Read-only bit | P = Programmable bit | U = Unimplemented bit, read as '0' |
| -n = Value when device is unprogrammed | | u = Unchanged from programmed state |

**MULLW** **Multiply Literal with W**

Syntax: MULLW k

Operands: $0 \le k \le 255$

Operation: (W) x k → PRODH:PRODL

Status Affected: None

Encoding:

| | | | |
|---|---|---|---|
| 0000 | 1101 | kkkk | kkkk |

Description: An unsigned multiplication is carried out between the contents of W and the 8-bit literal 'k'. The 16-bit result is placed in the PRODH:PRODL register pair. PRODH contains the high byte.

W is unchanged.

None of the Status flags are affected.

Note that neither overflow nor carry is possible in this operation. A zero result is possible but not detected.

Words: 1

Cycles: 1

Q Cycle Activity:

| Q1 | Q2 | Q3 | Q4 |
|---|---|---|---|
| Decode | Read literal 'k' | Process Data | Write registers PRODH: PRODL |

Example: MULLW 0C4h

Before Instruction
W = E2h
PRODH = ?
PRODL = ?

After Instruction
W = E2h
PRODH = ADh
PRODL = 08h

**MULWF** **Multiply W with f**

Syntax: MULWF f {,a}

Operands: $0 \le f \le 255$
$a \in [0,1]$

Operation: (W) x (f) → PRODH:PRODL

Status Affected: None

Encoding:

| | | | |
|---|---|---|---|
| 0000 | 001a | ffff | ffff |

Description: An unsigned multiplication is carried out between the contents of W and the register file location 'f'. The 16-bit result is stored in the PRODH:PRODL register pair. PRODH contains the high byte. Both W and 'f' are unchanged.

None of the Status flags are affected.

Note that neither overflow nor carry is possible in this operation. A zero result is possible but not detected.

If 'a' is '0', the Access Bank is selected. If 'a' is '1', the BSR is used to select the GPR bank (default).

If 'a' is '0' and the extended instruction set is enabled, this instruction operates in Indexed Literal Offset Addressing mode whenever $f \le 95$ (5Fh). See **Section 25.2.3 "Byte-Oriented and Bit-Oriented Instructions in Indexed Literal Offset Mode"** for details.

Words: 1

Cycles: 1

Q Cycle Activity:

| Q1 | Q2 | Q3 | Q4 |
|---|---|---|---|
| Decode | Read register 'f' | Process Data | Write registers PRODH: PRODL |

Example: MULWF REG, 1

Before Instruction
W = C4h
REG = B5h
PRODH = ?
PRODL = ?

After Instruction
W = C4h
REG = B5h
PRODH = 8Ah
PRODL = 94h

**RETFIE** **Return from Interrupt**

Syntax: RETFIE {s}

Operands: s ∈ [0,1]

Operation: (TOS) → PC,
1 → GIE/GIEH or PEIE/GIEL,
if s = 1
(WS) → W,
(STATUSS) → STATUS,
(BSRS) → BSR,
PCLATU, PCLATH are unchanged.

Status Affected: GIE/GIEH, PEIE/GIEL.

Encoding:

| | | | |
|---|---|---|---|
| 0000 | 0000 | 0001 | 000s |

Description: Return from Interrupt. Stack is popped and Top-of-Stack (TOS) is loaded into the PC. Interrupts are enabled by setting either the high or low priority global interrupt enable bit. If 's' = 1, the contents of the shadow registers, WS, STATUSS and BSRS, are loaded into their corresponding registers, W, STATUS and BSR. If 's' = 0, no update of these registers occurs (default).

Words: 1

Cycles: 2

Q Cycle Activity:

| Q1 | Q2 | Q3 | Q4 |
|---|---|---|---|
| Decode | No operation | No operation | POP PC from stack<br>Set GIEH or GIEL |
| No operation | No operation | No operation | No operation |

Example: RETFIE 1

After Interrupt
PC = TOS
W = WS
BSR = BSRS
STATUS = STATUSS
GIE/GIEH, PEIE/GIEL = 1

**RETLW** **Return Literal to W**

Syntax: RETLW k

Operands: 0 ≤ k ≤ 255

Operation: k → W,
(TOS) → PC,
PCLATU, PCLATH are unchanged

Status Affected: None

Encoding:

| | | | |
|---|---|---|---|
| 0000 | 1100 | kkkk | kkkk |

Description: W is loaded with the eight-bit literal 'k'. The program counter is loaded from the top of the stack (the return address). The high address latch (PCLATH) remains unchanged.

Words: 1

Cycles: 2

Q Cycle Activity:

| Q1 | Q2 | Q3 | Q4 |
|---|---|---|---|
| Decode | Read literal 'k' | Process Data | POP PC from stack, Write to W |
| No operation | No operation | No operation | No operation |

Example:

```
    CALL  TABLE  ; W contains table
                 ; offset value
                 ; W now has
                 ; table value
   :
TABLE
    ADDWF PCL    ; W = offset
    RETLW k0     ; Begin table
    RETLW k1     ;
   :
   :
    RETLW kn     ; End of table
```

Before Instruction
W = 07h
After Instruction
W = value of kn

## 27.3 DC Characteristics: PIC18F2585/2680/4585/4680 (Industrial) PIC18LF2585/2680/4585/4680 (Industrial)

| DC CHARACTERISTICS | | | Standard Operating Conditions (unless otherwise stated)<br>Operating temperature -40°C ≤ TA ≤ +85°C for industrial | | | |
|---|---|---|---|---|---|---|
| **Param No.** | **Symbol** | **Characteristic** | **Min** | **Max** | **Units** | **Conditions** |
| | VIL | **Input Low Voltage** | | | | |
| | | I/O ports: | | | | |
| D030 | | with TTL buffer | VSS | 0.15 VDD | V | VDD < 4.5V |
| D030A | | | — | 0.8 | V | 4.5V ≤ VDD ≤ 5.5V |
| D031 | | with Schmitt Trigger buffer | VSS | 0.2 VDD | V | |
| D031A | | RC3 and RC4 | VSS | 0.3 VDD | V | I²C™ enabled |
| D031B | | | VSS | 0.8 | V | SMBus enabled |
| D032 | | $\overline{MCLR}$ | VSS | 0.2 VDD | V | |
| D033 | | OSC1 | VSS | 0.3 VDD | V | HS, HSPLL modes |
| D033A | | OSC1 | VSS | 0.2 VDD | V | RC, EC modes(1) |
| D033B | | OSC1 | VSS | 0.3 | V | XT, LP modes |
| D034 | | T13CKI | VSS | 0.3 | V | |
| | VIH | **Input High Voltage** | | | | |
| | | I/O ports: | | | | |
| D040 | | with TTL buffer | 0.25 VDD + 0.8V | VDD | V | VDD < 4.5V |
| D040A | | | 2.0 | VDD | V | 4.5V ≤ VDD ≤ 5.5V |
| D041 | | with Schmitt Trigger buffer | 0.8 VDD | VDD | V | |
| D041A | | RC3 and RC4 | 0.7 VDD | VDD | V | I²C™ enabled |
| D041B | | | 2.1 | VDD | V | SMBus enabled |
| D042 | | $\overline{MCLR}$ | 0.8 VDD | VDD | V | |
| D043 | | OSC1 | 0.7 VDD | VDD | V | HS, HSPLL modes |
| D043A | | OSC1 | 0.8 VDD | VDD | V | EC mode |
| D043B | | OSC1 | 0.9 VDD | VDD | V | RC mode(1) |
| D043C | | OSC1 | 1.6 | VDD | V | XT, LP modes |
| D044 | | T13CKI | 1.6 | VDD | V | |
| | IIL | **Input Leakage Current(2,3)** | | | | |
| D060 | | I/O ports | — | ±1 | µA | VSS ≤ VPIN ≤ VDD,<br>Pin at high-impedance |
| D061 | | $\overline{MCLR}$ | — | ±5 | µA | Vss ≤ VPIN ≤ VDD |
| D063 | | OSC1 | — | ±5 | µA | Vss ≤ VPIN ≤ VDD |
| | IPU | **Weak Pull-up Current** | | | | |
| D070 | IPURB | PORTB weak pull-up current | 50 | 400 | µA | VDD = 5V, VPIN = VSS |

**Note 1:** In RC oscillator configuration, the OSC1/CLKI pin is a Schmitt Trigger input. It is not recommended that the PIC® device be driven with an external clock while in RC mode.

**2:** 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.

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

**4:** Parameter is characterized but not tested.

27.4.3 TIMING DIAGRAMS AND SPECIFICATIONS

**FIGURE 27-5: EXTERNAL CLOCK TIMING (ALL MODES EXCEPT PLL)**

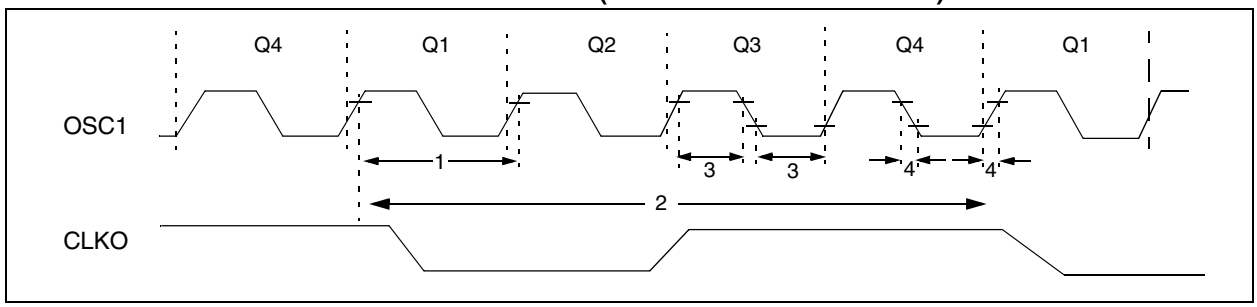
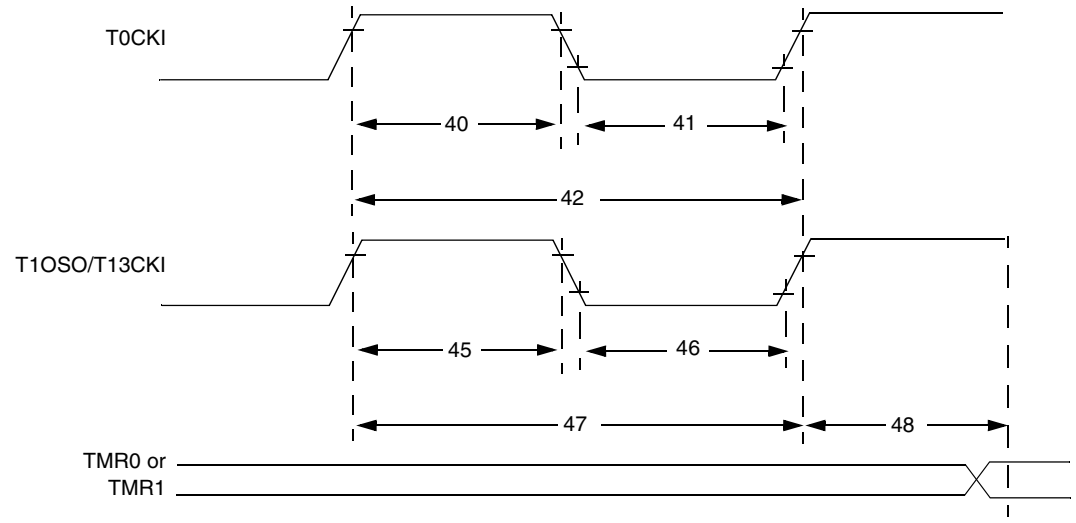



**TABLE 27-6: EXTERNAL CLOCK TIMING REQUIREMENTS**

| Param. No. | Symbol | Characteristic | Min | Max | Units | Conditions |
|---|---|---|---|---|---|---|
| 1A | FOSC | External CLKI Frequency(1) | DC | 1 | MHz | XT, RC Oscillator modes |
| | | | DC | 25 | MHz | HS Oscillator mode |
| | | | DC | 31.25 | kHz | LP Oscillator mode |
| | | | DC | 40 | MHz | EC Oscillator mode |
| | | Oscillator Frequency(1) | DC | 4 | MHz | RC Oscillator mode |
| | | | 0.1 | 4 | MHz | XT Oscillator mode |
| | | | 4 | 25 | MHz | HS Oscillator mode |
| | | | 4 | 10 | MHz | HSPLL Oscillator mode |
| | | | 5 | 200 | kHz | LP Oscillator mode |
| 1 | TOSC | External CLKI Period(1) | 1000 | — | ns | XT, RC Oscillator modes |
| | | | 40 | — | ns | HS Oscillator mode |
| | | | 32 | — | µs | LP Oscillator mode |
| | | | 25 | — | ns | EC Oscillator mode |
| | | Oscillator Period(1) | 250 | — | ns | RC Oscillator mode |
| | | | 250 | 1 | µs | XT Oscillator mode |
| | | | 40 | 250 | ns | HS Oscillator mode |
| | | | 100 | 250 | ns | HSPLL Oscillator mode |
| | | | 5 | 200 | µs | LP Oscillator mode |
| 2 | TCY | Instruction Cycle Time(1) | 100 | — | ns | TCY = 4/FOSC |
| 3 | TOSL, TOSH | External Clock in (OSC1) High or Low Time | 30 | — | ns | XT Oscillator mode |
| | | | 2.5 | — | µs | LP Oscillator mode |
| | | | 10 | — | ns | HS Oscillator mode |
| 4 | TOSR, TOSF | External Clock in (OSC1) Rise or Fall Time | — | 20 | ns | XT Oscillator mode |
| | | | — | 50 | ns | LP Oscillator mode |
| | | | — | 7.5 | ns | HS Oscillator mode |

**Note 1:** Instruction cycle period (TCY) equals four times the input oscillator time base period for all configurations except PLL. All specified values are based on characterization data for that particular oscillator type under standard operating conditions with the device executing code. Exceeding these specified limits may result in an unstable oscillator operation and/or higher than expected current consumption. All devices are tested to operate at “min.” values with an external clock applied to the OSC1/CLKI pin. When an external clock input is used, the “max.” cycle time limit is “DC” (no clock) for all devices.

**FIGURE 27-9: TIMER0 AND TIMER1 EXTERNAL CLOCK TIMINGS**




**Note:** Refer to Figure 27-4 for load conditions.

**TABLE 27-11: TIMER0 AND TIMER1 EXTERNAL CLOCK REQUIREMENTS**

| Param No. | Sym | Characteristic | | | Min | Max | Units | Conditions |
|---|---|---|---|---|---|---|---|---|
| 40 | TT0H | T0CKI High Pulse Width | | No prescaler | 0.5 TCY + 20 | — | ns | |
| | | | | With prescaler | 10 | — | ns | |
| 41 | TT0L | T0CKI Low Pulse Width | | No prescaler | 0.5 TCY + 20 | — | ns | |
| | | | | With prescaler | 10 | — | ns | |
| 42 | TT0P | T0CKI Period | | No prescaler | TCY + 10 | — | ns | |
| | | | | With prescaler | Greater of: 20 ns or (TCY + 40)/N | — | ns | N = prescale value (1, 2, 4,..., 256) |
| 45 | TT1H | T13CKI High Time | Synchronous, no prescaler | | 0.5 TCY + 20 | — | ns | |
| | | | Synchronous, with prescaler | PIC18**F**XXXX | 10 | — | ns | |
| | | | | PIC18**LF**XXXX | 25 | — | ns | VDD = 2.0V |
| | | | Asynchronous | PIC18**F**XXXX | 30 | — | ns | |
| | | | | PIC18**LF**XXXX | 50 | — | ns | VDD = 2.0V |
| 46 | TT1L | T13CKI Low Time | Synchronous, no prescaler | | 0.5 TCY + 5 | — | ns | |
| | | | Synchronous, with prescaler | PIC18**F**XXXX | 10 | — | ns | |
| | | | | PIC18**LF**XXXX | 25 | — | ns | VDD = 2.0V |
| | | | Asynchronous | PIC18**F**XXXX | 30 | — | ns | |
| | | | | PIC18**LF**XXXX | 50 | — | ns | VDD = 2.0V |
| 47 | TT1P | T13CKI Input Period | Synchronous | | Greater of: 20 ns or (TCY + 40)/N | — | ns | N = prescale value (1, 2, 4, 8) |
| | | | Asynchronous | | 60 | — | ns | |
| | FT1 | T13CKI Oscillator Input Frequency Range | | | DC | 50 | kHz | |
| 48 | TCKE2TMRI | Delay from External T13CKI Clock Edge to Timer Increment | | | 2 TOSC | 7 TOSC | — | |