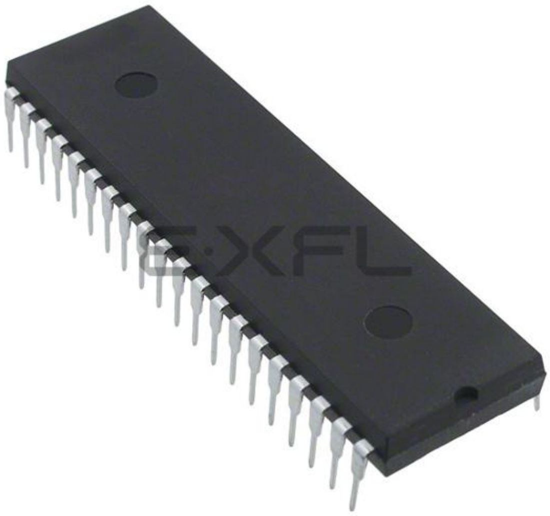


Welcome to E-XFL.COM

## 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 | 64MHz |
| Connectivity | I²C, LINbus, SPI, UART/USART |
| Peripherals | Brown-out Detect/Reset, LVD, POR, PWM, WDT |
| Number of I/O | 36 |
| Program Memory Size | 128KB (64K x 16) |
| Program Memory Type | FLASH |
| EEPROM Size | 1K x 8 |
| RAM Size | 3.6K x 8 |
| Voltage - Supply (Vcc/Vdd) | 1.8V ~ 3.6V |
| Data Converters | A/D 35x10b; D/A 1x5b |
| Oscillator Type | Internal |
| Operating Temperature | -40°C ~ 125°C (TA) |
| Mounting Type | Through Hole |
| Package / Case | 40-DIP (0.600", 15.24mm) |
| Supplier Device Package | 40-PDIP |
| Purchase URL | https://www.e-xfl.com/product-detail/microchip-technology/pic18lf47k40-e-p |

# 3.0 DEVICE CONFIGURATION

Device configuration consists of Configuration Words, Code Protection, Device ID and Rev ID.

## 3.1 Configuration Words

There are six Configuration Word bits that allow the user to setup the device with several choices of oscillators, Resets and memory protection options. These are implemented as Configuration Word 1 through Configuration Word 6 at 300000h through 30000Bh.

| | |
|---|---|
| **Note:** | The $\overline{\text{DEBUG}}$ bit in Configuration Words is managed automatically by device development tools including debuggers and programmers. For normal device operation, this bit should be maintained as a '1'. |

# 4.0 OSCILLATOR MODULE (WITH FAIL-SAFE CLOCK MONITOR)

## 4.1 Overview

The oscillator module has multiple clock sources and selection features that allow it to be used in a wide range of applications while maximizing performance and minimizing power consumption. Figure 4-1 illustrates a block diagram of the oscillator module.

Clock sources can be supplied from external oscillators, quartz-crystal resonators and ceramic resonators. In addition, the system clock source can be supplied from one of two internal oscillators and PLL circuits, with a choice of speeds selectable via software. Additional clock features include:

- Selectable system clock source between external or internal sources via software.
- Fail-Safe Clock Monitor (FSCM) designed to detect a failure of the external clock source (LP, XT, HS, ECH, ECM, ECL) and switch automatically to the internal oscillator.
- Oscillator Start-up Timer (OST) ensures stability of crystal oscillator sources.

The RSTOSC bits of Configuration Word 1 (Register 3-1) determine the type of oscillator that will be used when the device runs after Reset, including when it is first powered up.

If an external clock source is selected, the FEXTOSC bits of Configuration Word 1 must be used in conjunction with the RSTOSC bits to select the External Clock mode.

The external oscillator module can be configured in one of the following clock modes, by setting the FEXTOSC<2:0> bits of Configuration Word 1:

1. ECL – External Clock Low-Power mode (below 100 kHz)
2. ECM – External Clock Medium Power mode (100 kHz to 8 MHz)
3. ECH – External Clock High-Power mode (above 8 MHz)
4. LP – 32 kHz Low-Power Crystal mode.
5. XT – Medium Gain Crystal or Ceramic Resonator Oscillator mode (between 100 kHz and 8 MHz)
6. HS – High Gain Crystal or Ceramic Resonator mode (above 4 MHz)

The ECH, ECM, and ECL Clock modes rely on an external logic level signal as the device clock source. The LP, XT, and HS Clock modes require an external crystal or resonator to be connected to the device. Each mode is optimized for a different frequency range. The internal oscillator block produces low and high-frequency clock sources, designated LFINTOSC and HFINTOSC. (see Internal Oscillator Block, Figure 4-1). Multiple device clock frequencies may be derived from these clock sources.

**REGISTER 7-3: PMD2: PMD CONTROL REGISTER 2**

| U-0 | R/W-0/0 | R/W-0/0 | U-0 | U-0 | R/W-0/0 | R/W-0/0 | R/W-0/0 |
|---|---|---|---|---|---|---|---|
| — | DACMD | ADCMD | — | — | CMP2MD | CMP1MD | ZCDMD(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 | q = Value depends on condition |

bit 7 **Unimplemented:** Read as ‘0’

bit 6 **DACMD:** Disable DAC bit
1 = DAC module disabled
0 = DAC module enabled

bit 5 **ADCMD:** Disable ADC bit
1 = ADC module disabled
0 = ADC module enabled

bit 4-3 **Unimplemented:** Read as ‘0’

bit 2 **CMP2MD:** Disable Comparator CMP2 bit
1 = CMP2 module disabled
0 = CMP2 module enabled

bit 1 **CMP1MD:** Disable Comparator CMP1 bit
1 = CMP1 module disabled
0 = CMP1 module enabled

bit 0 **ZCDMD:** Disable Zero-Cross Detect module bit(1)
1 = ZCD module disabled
0 = ZCD module enabled

**Note 1:** Subject to $\overline{\text{ZCD}}$ bit in CONFIG2H.

### REGISTER 13-14: SCANLADRL: SCAN LOW 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 |
|---|---|---|---|---|---|---|---|
| LADR<7:0>(1, 2) | | | | | | | |
| 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 **LADR<7:0>**: Scan Start/Current Address bits(1, 2)
Least Significant bits of the current address to be fetched from, value increments on each fetch of memory

**Note 1:** Registers SCANLADRU/H/L form a 22-bit value, but are not guarded for atomic or asynchronous access; registers should only be read or written while SCANGO = `0` (SCANCON0 register).
**2:** While SCANGO = `1` (SCANCON0 register), writing to this register is ignored.

### REGISTER 13-15: SCANHADRU: SCAN HIGH ADDRESS UPPER BYTE REGISTER

| U-0 | U-0 | R/W-1/1 | R/W-1/1 | R/W-1/1 | R/W-1/1 | R/W-1/1 | R/W-1/1 |
|---|---|---|---|---|---|---|---|
| — | — | HADR<21:16> | | | | | |
| 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 **HADR<21:16>:** Scan End Address bits(1, 2)
Upper bits of the address at the end of the designated scan

**Note 1:** Registers SCANHADRU/H/L form a 22-bit value but are not guarded for atomic or asynchronous access; registers should only be read or written while SCANGO = `0` (SCANCON0 register).
**2:** While SCANGO = `1` (SCANCON0 register), writing to this register is ignored.

### REGISTER 14-24: IPR6: PERIPHERAL INTERRUPT PRIORITY REGISTER 6

| U-0 | U-0 | U-0 | U-0 | U-0 | U-0 | R/W-1/1 | R/W-1/1 |
|---|---|---|---|---|---|---|---|
| — | — | — | — | — | — | CCP2IP | CCP1IP |
| 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 7-2 **Unimplemented:** Read as ‘0’

bit 1 **CCP2IP:** ECCP2 Interrupt Priority bit
1 = High priority
0 = Low priority

bit 0 **CCP1IP:** ECCP1 Interrupt Priority bit
1 = High priority
0 = Low priority

### REGISTER 15-6: ODCONx: OPEN-DRAIN CONTROL REGISTER

| R/W-0/0 | R/W-0/0 | R/W-0/0 | R/W-0/0 | R/W-0/0 | R/W-0/0 | R/W-0/0 | R/W-0/0 |
|---|---|---|---|---|---|---|---|
| ODCx7 | ODCx6 | ODCx5 | ODCx4 | ODCx3 | ODCx2 | ODCx1 | ODCx0 |
| bit 7 | | | | | | | bit 0 |

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

bit 7-0 **ODCx<7:0>**: Open-Drain Configuration on Pins Rx<7:0>
`1` = Output drives only low-going signals (sink current only)
`0` = Output drives both high-going and low-going signals (source and sink current)

### TABLE 15-7: OPEN-DRAIN CONTROL REGISTERS

| Name | Device | | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|---|---|---|---|---|---|---|---|---|---|---|
| | 28 Pins | 40/44 Pins | | | | | | | | |
| ODCONA | X | X | ODCA7 | ODCA6 | ODCA5 | ODCA4 | ODCA3 | ODCA2 | ODCA1 | ODCA0 |
| ODCONB | X | X | ODCB7 | ODCB6 | ODCB5 | ODCB4 | ODCB3 | ODCB2 | ODCB1 | ODCB0 |
| ODCONC | X | X | ODCC7 | ODCC6 | ODCC5 | ODCC4 | ODCC3 | ODCC2 | ODCC1 | ODCC0 |
| ODCOND | X | | — | — | — | — | — | — | — | — |
| | | X | ODCD7 | ODCD6 | ODCD5 | ODCD4 | ODCD3 | ODCD2 | ODCD1 | ODCD0 |
| ODCONE | X | | — | — | — | — | — | — | — | — |
| | | X | — | — | — | — | — | ODCE2 | ODCE1 | ODCE0 |

**REGISTER 19-2: TxGCON: TIMERx GATE CONTROL REGISTER**

| R/W-0/u | R/W-0/u | R/W-0/u | R/W-0/u | R/W-0/u | R-x | U-0 | U-0 |
|---|---|---|---|---|---|---|---|
| GE | GPOL | GTM | GSPM | GGO/$\overline{\text{DONE}}$ | GVAL | — | — |
| 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 **GE:** Timerx Gate Enable bit
If TMRxON = 1:
1 = Timerx counting is controlled by the Timerx gate function
0 = Timerx is always counting
If TMRxON = 0:
This bit is ignored

bit 6 **GPOL:** Timerx Gate Polarity bit
1 = Timerx gate is active-high (Timerx counts when gate is high)
0 = Timerx gate is active-low (Timerx counts when gate is low)

bit 5 **GTM:** Timerx Gate Toggle Mode bit
1 = Timerx Gate Toggle mode is enabled
0 = Timerx Gate Toggle mode is disabled and Toggle flip-flop is cleared
Timerx Gate Flip Flop Toggles on every rising edge

bit 4 **GSPM:** Timerx Gate Single Pulse Mode bit
1 = Timerx Gate Single Pulse mode is enabled and is controlling Timerx gate)
0 = Timerx Gate Single Pulse mode is disabled

bit 3 **GGO/$\overline{\text{DONE}}$:** Timerx Gate Single Pulse Acquisition Status bit
1 = Timerx Gate Single Pulse Acquisition is ready, waiting for an edge
0 = Timerx Gate Single Pulse Acquisition has completed or has not been started.
This bit is automatically cleared when TxGSPM is cleared.

bit 2 **GVAL:** Timerx Gate Current State bit
Indicates the current state of the Timerx gate that could be provided to TMRxH:TMRxL
Unaffected by Timerx Gate Enable (TMRxGE)

bit 1-0 **Unimplemented:** Read as ‘0’

### 20.5.6 EDGE-TRIGGERED ONE-SHOT MODE

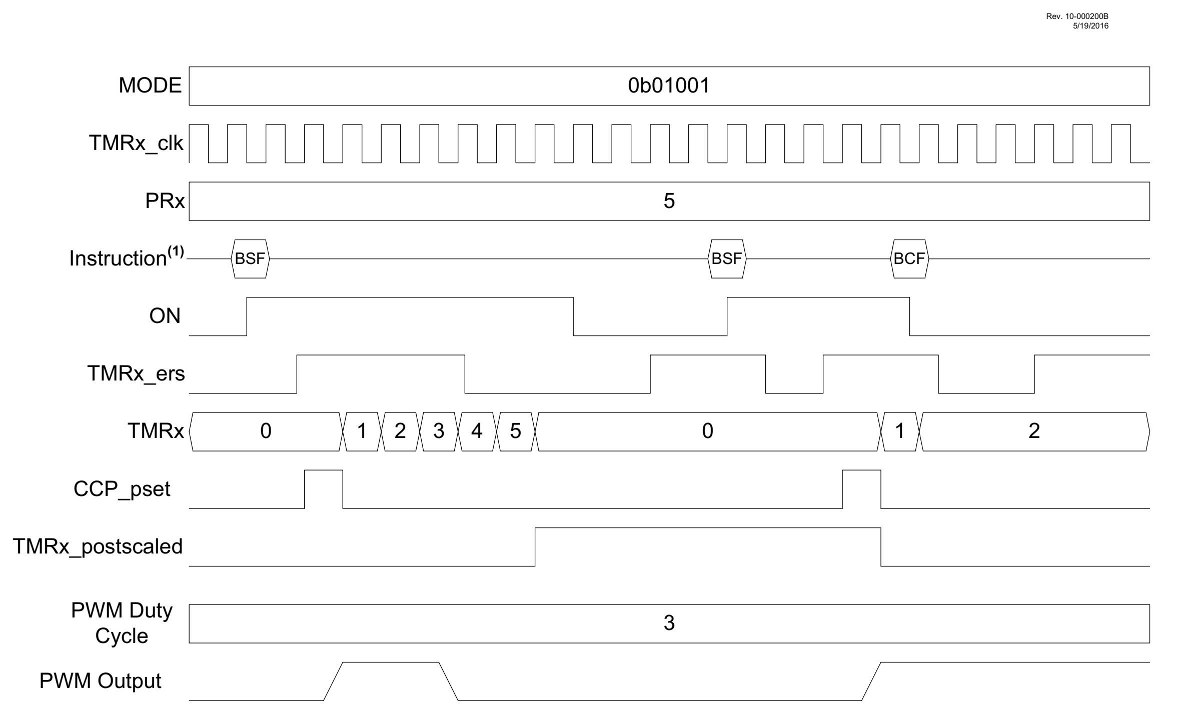
The Edge-Triggered One-Shot modes start the timer on an edge from the external signal input, after the ON bit is set, and clear the ON bit when the timer matches the PRx period value. The following edges will start the timer:

- Rising edge (MODE<4:0> = `01001`)
- Falling edge (MODE<4:0> = `01010`)
- Rising or Falling edge (MODE<4:0> = `01011`)

If the timer is halted by clearing the ON bit then another TMRx_ers edge is required after the ON bit is set to resume counting. Figure 20-9 illustrates operation in the rising edge One-Shot mode.

When Edge-Triggered One-Shot mode is used in conjunction with the CCP then the edge-trigger will activate the PWM drive and the PWM drive will deactivate when the timer matches the CCPRx pulse width value and stay deactivated when the timer halts at the PRx period count match.

**FIGURE 20-9: EDGE-TRIGGERED ONE-SHOT MODE TIMING DIAGRAM (MODE = `01001`)**




Note 1: BSF and BCF represent Bit-Set File and Bit-Clear File instructions executed by the CPU to set or clear the ON bit of TxCON. CPU execution is asynchronous to the timer clock input.

### 20.5.7 EDGE-TRIGGERED HARDWARE LIMIT ONE-SHOT MODE

In Edge-Triggered Hardware Limit One-Shot modes the timer starts on the first external signal edge after the ON bit is set and resets on all subsequent edges. Only the first edge after the ON bit is set is needed to start the timer. The counter will resume counting automatically two clocks after all subsequent external Reset edges. Edge triggers are as follows:

- Rising edge start and Reset (MODE<4:0> = `01100`)
- Falling edge start and Reset (MODE<4:0> = `01101`)

The timer resets and clears the ON bit when the timer value matches the PRx period value. External signal edges will have no effect until after software sets the ON bit. Figure 20-10 illustrates the rising edge hardware limit one-shot operation.

When this mode is used in conjunction with the CCP then the first starting edge trigger, and all subsequent Reset edges, will activate the PWM drive. The PWM drive will deactivate when the timer matches the CCPRx pulse-width value and stay deactivated until the timer halts at the PRx period match unless an external signal edge resets the timer before the match occurs.

## 22.0 PULSE-WIDTH MODULATION (PWM )

The PWM module generates a Pulse-Width Modulated signal determined by the duty cycle, period, and resolution that are configured by the following registers:

- PRx
- TxCON
- PWMxDCH
- PWMxDCL
- PWMxCON

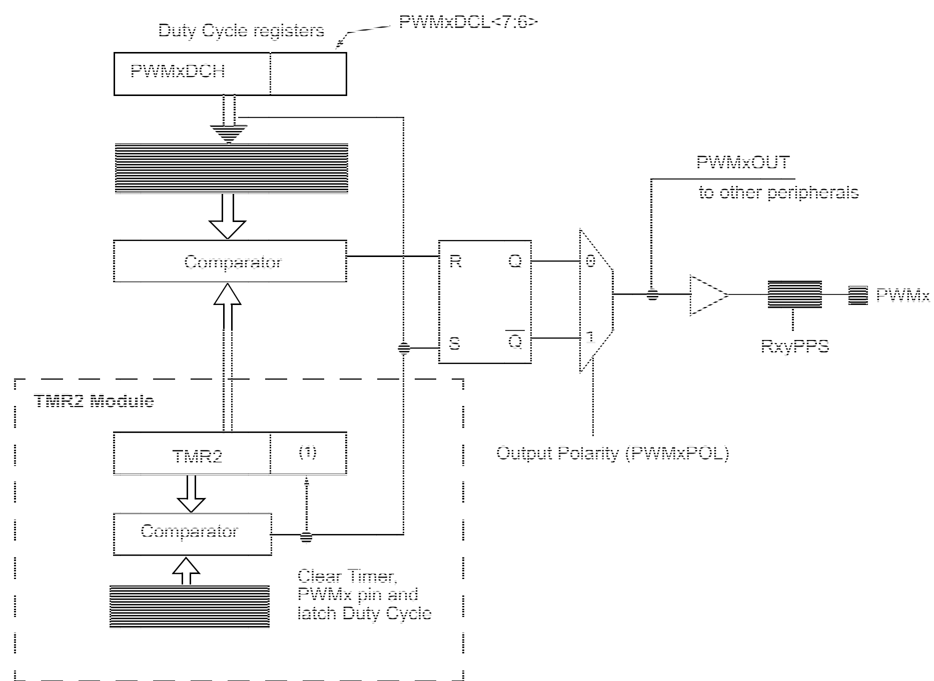
| Note: | The corresponding TRIS bit must be cleared to enable the PWM output on the PWMx pin. |
|---|---|

Each PWM module can select the timer source that controls the module. Each module has an independent timer selection which can be accessed using the CCPTMRS register (Register 21-2). Please note that the PWM mode operation is described with respect to TMR2 in the following sections.

Figure 22-1 shows a simplified block diagram of PWM operation.

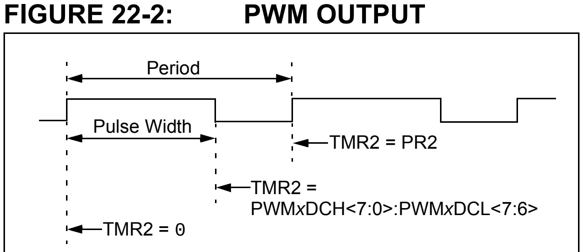
Figure 22-2 shows a typical waveform of the PWM signal.

**FIGURE 22-1: SIMPLIFIED PWM BLOCK DIAGRAM**




**Note 1:** 8-bit timer is concatenated with the two Least Significant bits of 1/Fosc adjusted by the Timer2 prescaler to create a 10-bit time base.

**FIGURE 22-2: PWM OUTPUT**




For a step-by-step procedure on how to set up this module for PWM operation, refer to **Section 22.1.9 “Setup for PWM Operation using PWMx Pins”**.

### 26.9.4 SLAVE MODE 10-BIT ADDRESS RECEPTION

This section describes a standard sequence of events for the MSSP module configured as an I$^2$C slave in 10-bit Addressing mode.

Figure 26-20 is used as a visual reference for this description.

This is a step by step process of what must be done by slave software to accomplish I$^2$C communication.

1. Bus starts Idle.
2. Master sends Start condition; S bit of SSPxSTAT is set; SSPxIF is set if interrupt on Start detect is enabled.
3. Master sends matching high address with R/$\overline{\text{W}}$ bit clear; UA bit of the SSPxSTAT register is set.
4. Slave sends $\overline{\text{ACK}}$ and SSPxIF is set.
5. Software clears the SSPxIF bit.
6. Software reads received address from SSPxBUF clearing the BF flag.
7. Slave loads low address into SSPxADD, releasing SCL.
8. Master sends matching low address byte to the slave; UA bit is set.

| **Note:** | Updates to the SSPxADD register are not allowed until after the $\overline{\text{ACK}}$ sequence. |
|---|---|

9. Slave sends $\overline{\text{ACK}}$ and SSPxIF is set.

| **Note:** | If the low address does not match, SSPxIF and UA are still set so that the slave software can set SSPxADD back to the high address. BF is not set because there is no match. CKP is unaffected. |
|---|---|

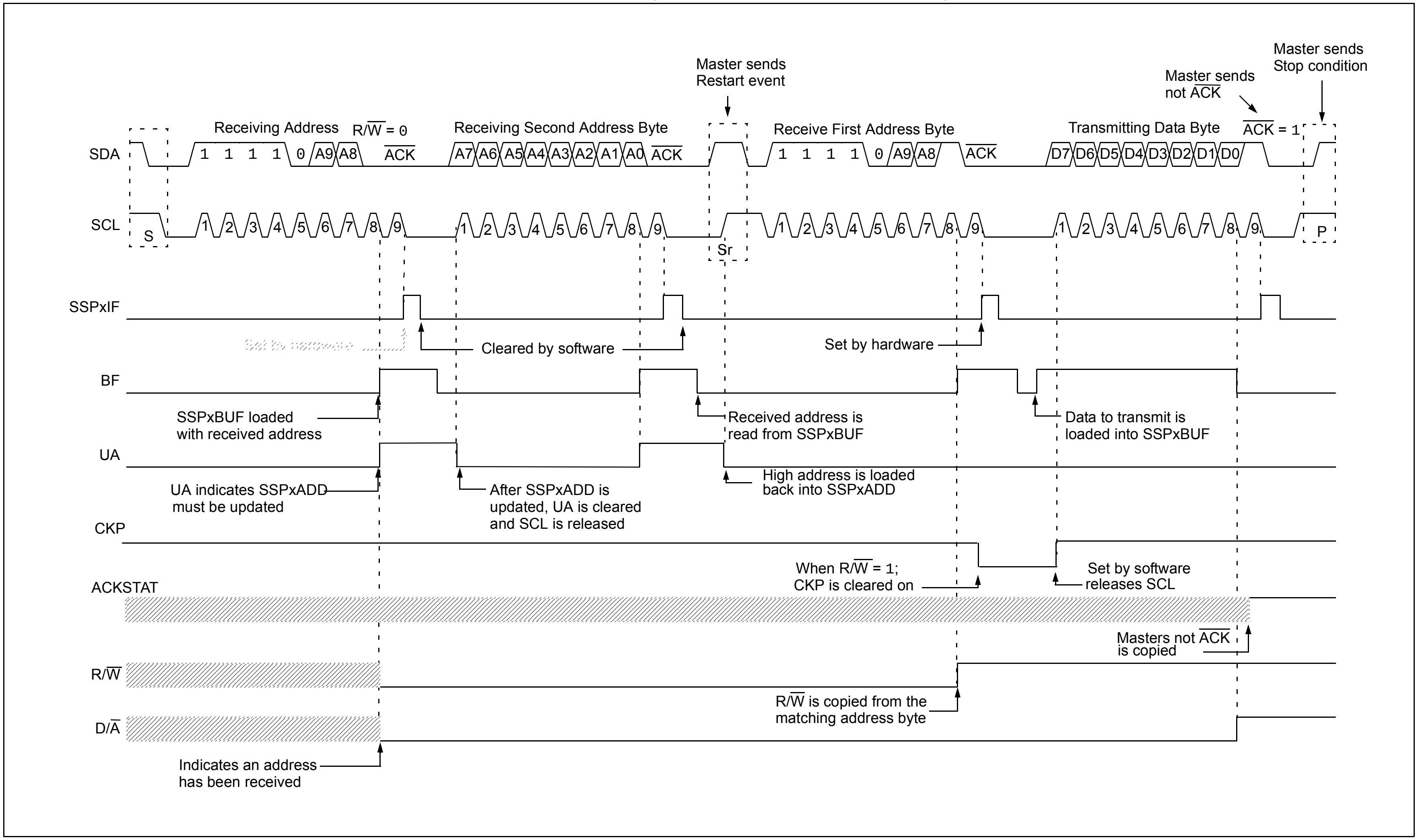
10. Slave clears SSPxIF.
11. Slave reads the received matching address from SSPxBUF clearing BF.
12. Slave loads high address into SSPxADD.
13. Master clocks a data byte to the slave and clocks out the slaves $\overline{\text{ACK}}$ on the ninth SCL pulse; SSPxIF is set.
14. If SEN bit of SSPxCON2 is set, CKP is cleared by hardware and the clock is stretched.
15. Slave clears SSPxIF.
16. Slave reads the received byte from SSPxBUF clearing BF.
17. If SEN is set the slave sets CKP to release the SCL.
18. Steps 13-17 repeat for each received byte.
19. Master sends Stop to end the transmission.

### 26.9.5 10-BIT ADDRESSING WITH ADDRESS OR DATA HOLD

Reception using 10-bit addressing with AHEN or DHEN set is the same as with 7-bit modes. The only difference is the need to update the SSPxADD register using the UA bit. All functionality, specifically when the CKP bit is cleared and SCL line is held low are the same. Figure 26-21 can be used as a reference of a slave in 10-bit addressing with AHEN set.

Figure 26-22 shows a standard waveform for a slave transmitter in 10-bit Addressing mode.

**FIGURE 26-22: I²C SLAVE, 10-BIT ADDRESS, TRANSMISSION (SEN = 0, AHEN = 0, DHEN = 0)**



### 26.9.6 CLOCK STRETCHING

Clock stretching occurs when a device on the bus holds the SCL line low, effectively pausing communication. The slave may stretch the clock to allow more time to handle data or prepare a response for the master device. A master device is not concerned with stretching as anytime it is active on the bus and not transferring data it is stretching. Any stretching done by a slave is invisible to the master software and handled by the hardware that generates SCL.

The CKP bit of the SSPxCON1 register is used to control stretching in software. Any time the CKP bit is cleared, the module will wait for the SCL line to go low and then hold it. Setting CKP will release SCL and allow more communication.

#### 26.9.6.1 Normal Clock Stretching

Following an $\overline{\text{ACK}}$ if the R/$\overline{\text{W}}$ bit of SSPxSTAT is set, a read request, the slave hardware will clear CKP. This allows the slave time to update SSPxBUF with data to transfer to the master. If the SEN bit of SSPxCON2 is set, the slave hardware will always stretch the clock after the $\overline{\text{ACK}}$ sequence. Once the slave is ready; CKP is set by software and communication resumes.

> **Note 1:** The BF bit has no effect on if the clock will be stretched or not. This is different than previous versions of the module that would not stretch the clock, clear CKP, if SSPxBUF was read before the ninth falling edge of SCL.
>
> **2:** Previous versions of the module did not stretch the clock for a transmission if SSPxBUF was loaded before the ninth falling edge of SCL. It is now always cleared for read requests.

#### 26.9.6.2 10-bit Addressing Mode

In 10-bit Addressing mode, when the UA bit is set, the clock is always stretched. This is the only time the SCL is stretched without CKP being cleared. SCL is released immediately after a write to SSPxADD.

> **Note:** Previous versions of the module did not stretch the clock if the second address byte did not match.

#### 26.9.6.3 Byte NACKing

When the AHEN bit of SSPxCON3 is set; CKP is cleared by hardware after the eighth falling edge of SCL for a received matching address byte. When the DHEN bit of SSPxCON3 is set; CKP is cleared after the eighth falling edge of SCL for received data.

Stretching after the eighth falling edge of SCL allows the slave to look at the received address or data and decide if it wants to ACK the received data.

### 26.9.7 CLOCK SYNCHRONIZATION AND THE CKP BIT

Any time the CKP bit is cleared, the module will wait for the SCL line to go low and then hold it. However, clearing the CKP bit will not assert the SCL output low until the SCL output is already sampled low. Therefore, the CKP bit will not assert the SCL line until an external I$^2$C master device has already asserted the SCL line. The SCL output will remain low until the CKP bit is set and all other devices on the I$^2$C bus have released SCL. This ensures that a write to the CKP bit will not violate the minimum high time requirement for SCL (see Figure 26-23).

**FIGURE 26-23: CLOCK SYNCHRONIZATION TIMING**

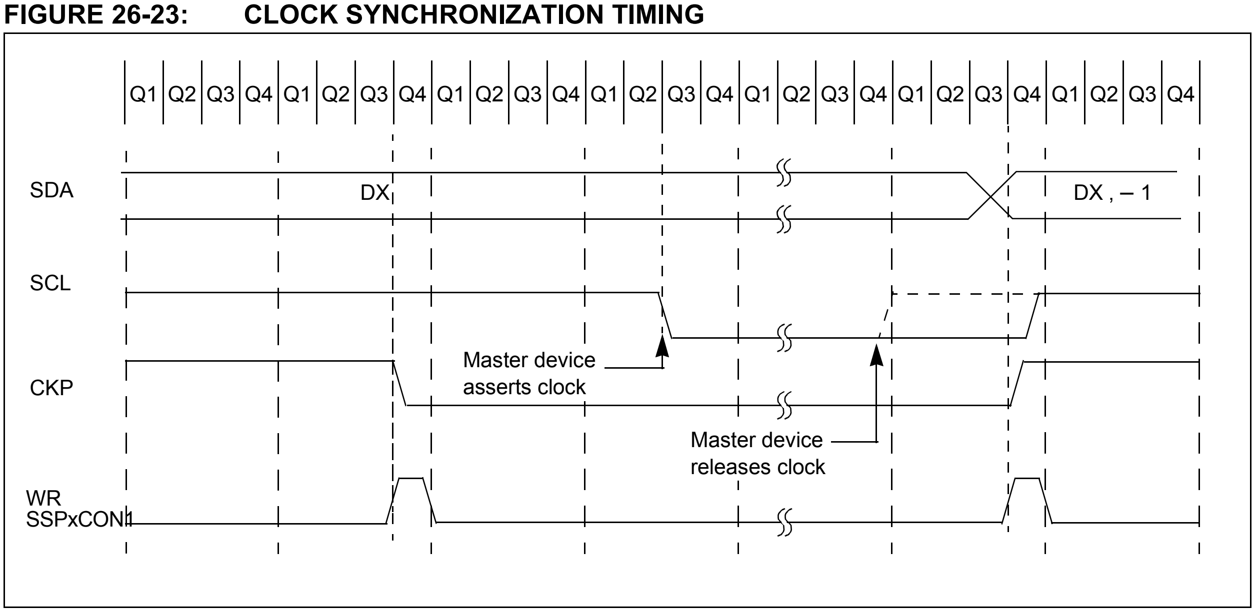


**TABLE 26-4: SUMMARY OF REGISTERS ASSOCIATED WITH I²C OPERATION**

| Name | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | Reset Values on Page |
|---|---|---|---|---|---|---|---|---|---|
| INTCON | GIE/GIEH | PEIE/GIEL | IPEN | — | — | INT2EDG | INT1EDG | INT0EDG | 170 |
| PIE3 | RC2IE | TX2IE | RC1IE | TX1IE | BCL2IE | SSP2IE | BCL1IE | SSP1IE | 182 |
| PIR3 | RC2IF | TX2IF | RC1IF | TX1IF | BCL2IF | SSP2IF | BCL1IF | SSP1IF | 174 |
| IPR3 | RC2IP | TX2IP | RC1IP | TX1IP | BCL2IP | SSP2IP | BCL1IP | SSP1IP | 190 |
| RxyPPS | — | — | — | RxyPPS<4:0> | | | | | 218 |
| SSPxADD | ADD<7:0> | | | | | | | | 340 |
| SSPxBUF | BUF<7:0> | | | | | | | | 336* |
| SSPxCLKPPS | — | — | — | SSPCLKPPS<4:0> | | | | | 216 |
| SSPxCON1 | WCOL | SSPOV | SSPEN | CKP | SSPM<3:0> | | | | 338 |
| SSPxCON2 | GCEN | ACKSTAT | ACKDT | ACKEN | RCEN | PEN | RSEN | SEN | 355 |
| SSPxCON3 | ACKTIM | PCIE | SCIE | BOEN | SDAHT | SBCDE | AHEN | DHEN | 339 |
| SSPxDATPPS | — | — | — | SSPDATPPS<4:0> | | | | | 216 |
| SSPxMSK | MSK<7:0> | | | | | | | | 357 |
| SSPxSTAT | SMP | CKE | $D/\overline{A}$ | P | S | $R/\overline{W}$ | UA | BF | 337 |

**Legend:** — = unimplemented location, read as ‘0’. Shaded cells are not used by the MSSP module in I²C mode.
* Page provides register information.

### 27.4.2 AUTO-BAUD OVERFLOW

During the course of automatic baud detection, the ABDOVF bit of the BAUDxCON register will be set if the baud rate counter overflows before the fifth rising edge is detected on the RXx pin. The ABDOVF bit indicates that the counter has exceeded the maximum count that can fit in the 16 bits of the SPxBRGH:SPxBRGL register pair. After the ABDOVF bit has been set, the counter continues to count until the fifth rising edge is detected on the RXx pin. Upon detecting the fifth RX edge, the hardware will set the RCxIF interrupt flag and clear the ABDEN bit of the BAUDxCON register. The RCxIF flag can be subsequently cleared by reading the RCxREG register. The ABDOVF flag of the BAUDxCON register can be cleared by software directly.

To terminate the auto-baud process before the RCxIF flag is set, clear the ABDEN bit then clear the ABDOVF bit of the BAUDxCON register. The ABDOVF bit will remain set if the ABDEN bit is not cleared first.

### 27.4.3 AUTO-WAKE-UP ON BREAK

During Sleep mode, all clocks to the EUSART are suspended. Because of this, the Baud Rate Generator is inactive and a proper character reception cannot be performed. The Auto-Wake-up feature allows the controller to wake-up due to activity on the RX/DT line. This feature is available only in Asynchronous mode.

The Auto-Wake-up feature is enabled by setting the WUE bit of the BAUDxCON register. Once set, the normal receive sequence on RX/DT is disabled, and the EUSART remains in an Idle state, monitoring for a wake-up event independent of the CPU mode. A wake-up event consists of a high-to-low transition on the RX/DT line. (This coincides with the start of a Sync Break or a wake-up signal character for the LIN protocol.)

The EUSART module generates an RCxIF interrupt coincident with the wake-up event. The interrupt is generated synchronously to the Q clocks in normal CPU operating modes (Figure 27-7), and asynchronously if the device is in Sleep mode (Figure 27-8). The interrupt condition is cleared by reading the RCxREG register.

The WUE bit is automatically cleared by the low-to-high transition on the RX line at the end of the Break. This signals to the user that the Break event is over. At this point, the EUSART module is in Idle mode waiting to receive the next character.

#### 27.4.3.1 Special Considerations

<u>Break Character</u>

To avoid character errors or character fragments during a wake-up event, the wake-up character must be all zeros.

When the wake-up is enabled the function works independent of the low time on the data stream. If the WUE bit is set and a valid non-zero character is received, the low time from the Start bit to the first rising edge will be interpreted as the wake-up event. The remaining bits in the character will be received as a fragmented character and subsequent characters can result in framing or overrun errors.

Therefore, the initial character in the transmission must be all '0's. This must be ten or more bit times, 13-bit times recommended for LIN bus, or any number of bit times for standard RS-232 devices.

<u>Oscillator Start-up Time</u>

Oscillator start-up time must be considered, especially in applications using oscillators with longer start-up intervals (i.e., LP, XT or HS/PLL mode). The Sync Break (or wake-up signal) character must be of sufficient length, and be followed by a sufficient interval, to allow enough time for the selected oscillator to start and provide proper initialization of the EUSART.

<u>WUE Bit</u>

The wake-up event causes a receive interrupt by setting the RCxIF bit. The WUE bit is cleared in hardware by a rising edge on RX/DT. The interrupt condition is then cleared in software by reading the RCxREG register and discarding its contents.

To ensure that no actual data is lost, check the RCIDL bit to verify that a receive operation is not in process before setting the WUE bit. If a receive operation is not occurring, the WUE bit may then be set just prior to entering the Sleep mode.

**TABLE 27-9: SUMMARY OF REGISTERS ASSOCIATED WITH SYNCHRONOUS SLAVE TRANSMISSION**

| Name | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | Register on Page |
|---|---|---|---|---|---|---|---|---|---|
| BAUDxCON | ABDOVF | RCIDL | — | SCKP | BRG16 | — | WUE | ABDEN | 395 |
| INTCON | GIE/GIEH | PEIE/GIEL | IPEN | — | — | INT2EDG | INT1EDG | INT0EDG | 170 |
| PIE3 | RC2IE | TX2IE | RC1IE | TX1IE | BCL2IE | SSP2IE | BCL1IE | SSP1IE | 182 |
| PIR3 | RC2IF | TX2IF | RC1IF | TX1IF | BCL2IF | SSP2IF | BCL1IF | SSP1IF | 174 |
| IPR3 | RC2IP | TX2IP | RC1IP | TX1IP | BCL2IP | SSP2IP | BCL1IP | SSP1IP | 190 |
| RCxSTA | SPEN | RX9 | SREN | CREN | ADDEN | FERR | OERR | RX9D | 394 |
| RxyPPS | — | — | — | RxyPPS<4:0> | | | | | 218 |
| TXxPPS | — | — | — | TXPPS<4:0> | | | | | 216 |
| TXxREG | EUSARTx Transmit Data Register | | | | | | | | 396* |
| TXxSTA | CSRC | TX9 | TXEN | SYNC | SENDB | BRGH | TRMT | TX9D | 393 |

**Legend:** — = unimplemented location, read as ‘0’. Shaded cells are not used for synchronous slave transmission.
* Page provides register information.

## 31.3 ADC Acquisition Requirements

For the ADC to meet its specified accuracy, the charge holding capacitor (CHOLD) must be allowed to fully charge to the input channel voltage level. The Analog Input model is shown in Figure 31-4. The source impedance (RS) and the internal sampling switch (RSS) impedance directly affect the time required to charge the capacitor CHOLD. The sampling switch (RSS) impedance varies over the device voltage (VDD), refer to Figure 31-4. **The maximum recommended impedance for analog sources is 10 kΩ.** As the source impedance is decreased, the acquisition time may be decreased. After the analog input channel is selected (or changed), an ADC acquisition must be completed before the conversion can be started. To calculate the minimum acquisition time, Equation 31-1 may be used. This equation assumes that 1/2 LSb error is used (1,024 steps for the ADC). The 1/2 LSb error is the maximum error allowed for the ADC to meet its specified resolution.

**EQUATION 31-1: ACQUISITION TIME EXAMPLE**

*Assumptions:* $Temperature = 50°C$ *and external impedance of* $10k\Omega$ $5.0V$ $V_{DD}$

$$\begin{aligned} T_{ACQ} &= \text{Amplifier Settling Time} + \text{Hold Capacitor Charging Time} + \text{Temperature Coefficient} \\ &= T_{AMP} + T_C + T_{COFF} \\ &= 2\mu s + T_C + [(Temperature - 25°C)(0.05\mu s/°C)] \end{aligned}$$

*The value for* $T_C$ *can be approximated with the following equations:*

$$V_{APPLIED}\left(1 - \frac{1}{(2^{n+1}) - 1}\right) = V_{CHOLD} \qquad ;[1]\ V_{CHOLD} \text{ charged to within 1/2 lsb}$$

$$V_{APPLIED}\left(1 - e^{\frac{-T_C}{RC}}\right) = V_{CHOLD} \qquad ;[2]\ V_{CHOLD} \text{ charge response to } V_{APPLIED}$$

$$V_{APPLIED}\left(1 - e^{\frac{-Tc}{RC}}\right) = V_{APPLIED}\left(1 - \frac{1}{(2^{n+1}) - 1}\right) \qquad ;\text{combining [1] and [2]}$$

*Note: Where n = number of bits of the ADC.*

*Solving for* $T_C$:

$$\begin{aligned} T_C &= -C_{HOLD}(R_{IC} + R_{SS} + R_S)\ ln(1/2047) \\ &= -10pF(1k\Omega + 7k\Omega + 10k\Omega)\ ln(0.0004885) \\ &= 1.37\mu s \end{aligned}$$

*Therefore:*

$$\begin{aligned} T_{ACQ} &= 2\mu s + 892ns + [(50°C - 25°C)(0.05\mu s/°C)] \\ &= 4.62\mu s \end{aligned}$$

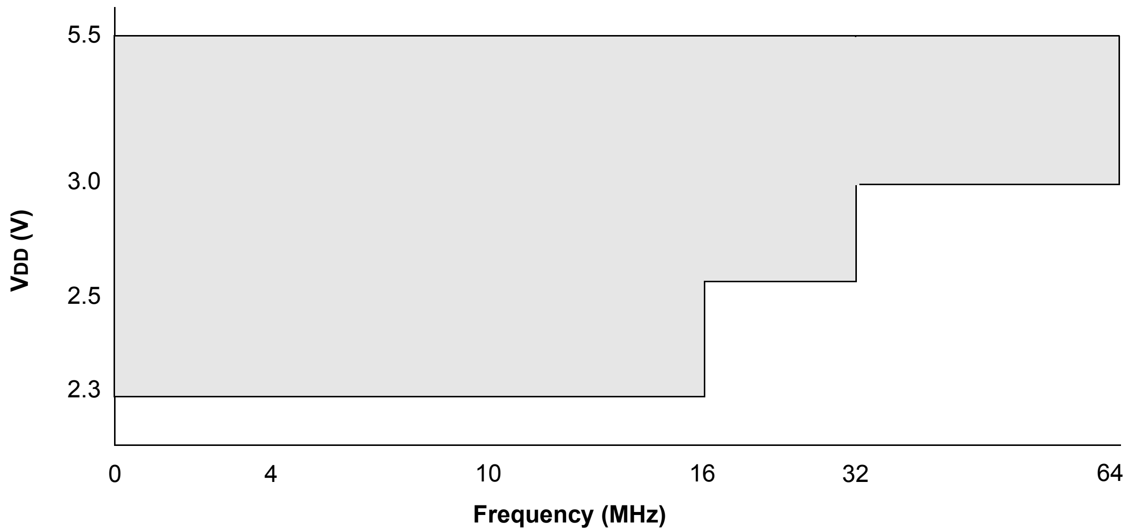
**Note 1:** The reference voltage (VREF) has no effect on the equation, since it cancels itself out.

**2:** The charge holding capacitor (CHOLD) is not discharged after each conversion.

**3:** The maximum recommended impedance for analog sources is 10 kΩ. This is required to meet the pin leakage specification.

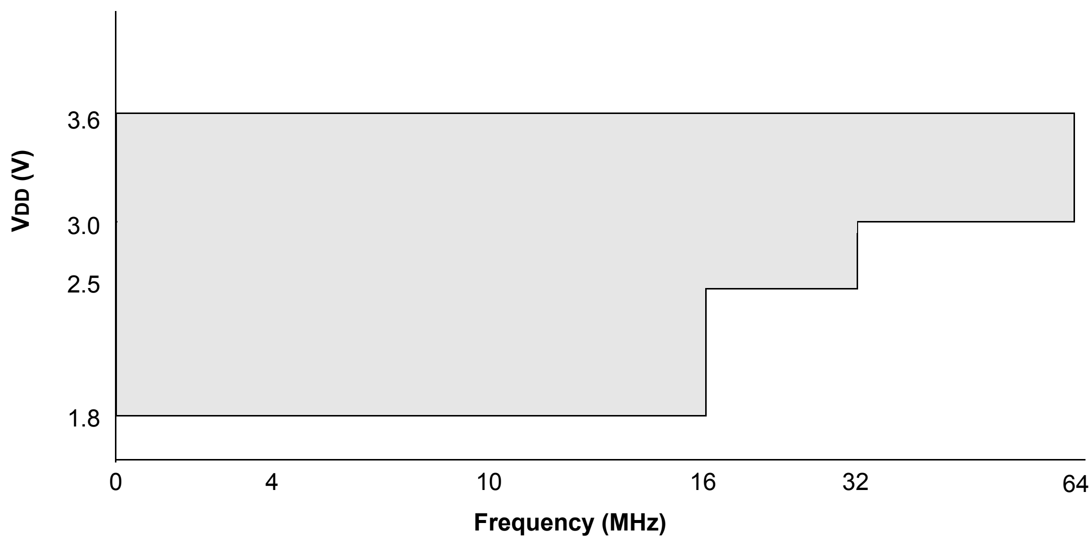
 

**FIGURE 37-1: VOLTAGE FREQUENCY GRAPH, -40°C $\le$ TA $\le$ +125°C, PIC18F27/47K40 ONLY**




**Note 1:** The shaded region indicates the permissible combinations of voltage and frequency.
**2:** Refer to Table 37-7 for each Oscillator mode's supported frequencies.

**FIGURE 37-2: VOLTAGE FREQUENCY GRAPH, -40°C $\le$ TA $\le$ +125°C, PIC18LF27/47K40 ONLY**




**Note 1:** The shaded region indicates the permissible combinations of voltage and frequency.
**2:** Refer to Table 37-7 for each Oscillator mode's supported frequencies.

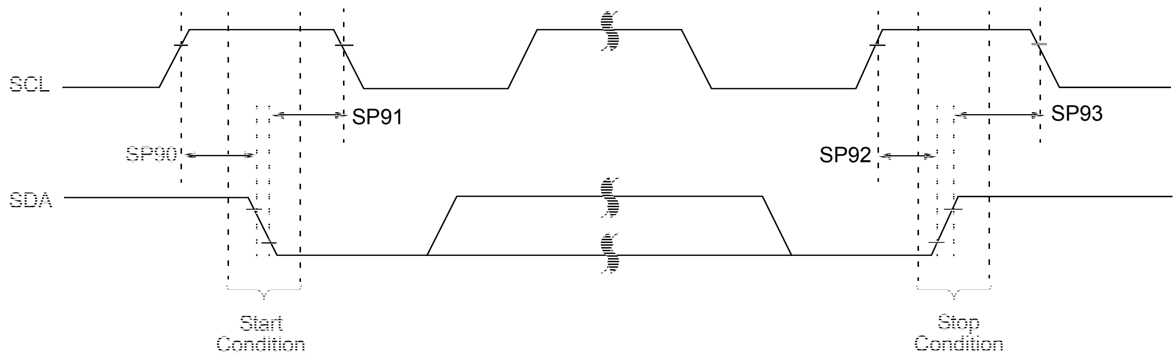
### TABLE 37-5: MEMORY PROGRAMMING SPECIFICATIONS

| Standard Operating Conditions (unless otherwise stated) | | | | | | | |
|---|---|---|---|---|---|---|---|
| **Param No.** | **Sym.** | **Characteristic** | **Min.** | **Typ†** | **Max.** | **Units** | **Conditions** |
| **Data EEPROM Memory Specifications** | | | | | | | |
| MEM20 | $E_D$ | DataEE Byte Endurance | 100k | — | — | E/W | -40°C ≤ TA ≤ +85°C |
| MEM21 | $T_{D\_RET}$ | Characteristic Retention | — | 40 | — | Year | Provided no other specifications are violated |
| MEM22 | $N_{D\_REF}$ | Total Erase/Write Cycles before Refresh | 1M<br>500k | 10M<br>— | —<br>— | E/W | -40°C ≤ TA ≤ +60°C<br>-40°C ≤ TA ≤ +85°C |
| MEM23 | $V_{D\_RW}$ | VDD for Read or Erase/Write operation | VDDMIN | — | VDDMAX | V | |
| MEM24 | $T_{D\_BEW}$ | Byte Erase and Write Cycle Time | — | 4.0 | 5.0 | ms | |
| **Program Flash Memory Specifications** | | | | | | | |
| MEM30 | $E_P$ | Flash Memory Cell Endurance | 10k | — | — | E/W | -40°C ≤ TA ≤ +85°C (**Note 1**) |
| MEM32 | $T_{P\_RET}$ | Characteristic Retention | — | 40 | — | Year | Provided no other specifications are violated |
| MEM33 | $V_{P\_RD}$ | VDD for Read operation | VDDMIN | — | VDDMAX | V | |
| MEM34 | $V_{P\_REW}$ | VDD for Row Erase or Write operation | VDDMIN | — | VDDMAX | V | |
| MEM35 | $T_{P\_REW}$ | Self-Timed Row Erase or Self-Timed Write | — | 2.0 | 2.5 | ms | |

† 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:** Flash Memory Cell Endurance for the Flash memory is defined as: One Row Erase operation and one Self-Timed Write.

**FIGURE 37-20: $I^2C$ BUS START/STOP BITS TIMING**



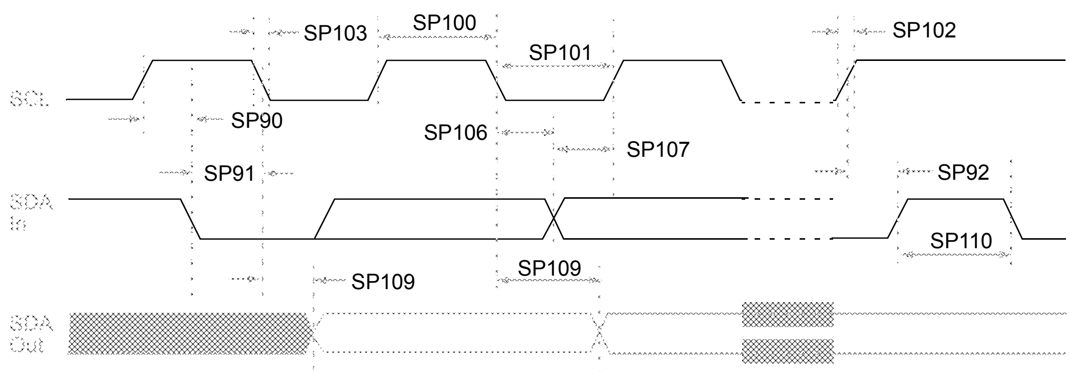

**Note**: Refer to Figure 37-4 for load conditions.

**TABLE 37-24: $I^2C$ BUS START/STOP BITS REQUIREMENTS**

| Standard Operating Conditions (unless otherwise stated) | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| **Param No.** | **Symbol** | **Characteristic** | | **Min.** | **Typ** | **Max.** | **Units** | **Conditions** |
| SP90* | TSU:STA | Start condition | 100 kHz mode | 4700 | — | — | ns | Only relevant for Repeated Start condition |
| | | Setup time | 400 kHz mode | 600 | — | — | | |
| SP91* | THD:STA | Start condition | 100 kHz mode | 4000 | — | — | ns | After this period, the first clock pulse is generated |
| | | Hold time | 400 kHz mode | 600 | — | — | | |
| SP92* | TSU:STO | Stop condition | 100 kHz mode | 4700 | — | — | ns | |
| | | Setup time | 400 kHz mode | 600 | — | — | | |
| SP93 | THD:STO | Stop condition | 100 kHz mode | 4000 | — | — | ns | |
| | | Hold time | 400 kHz mode | 600 | — | — | | |

* These parameters are characterized but not tested.

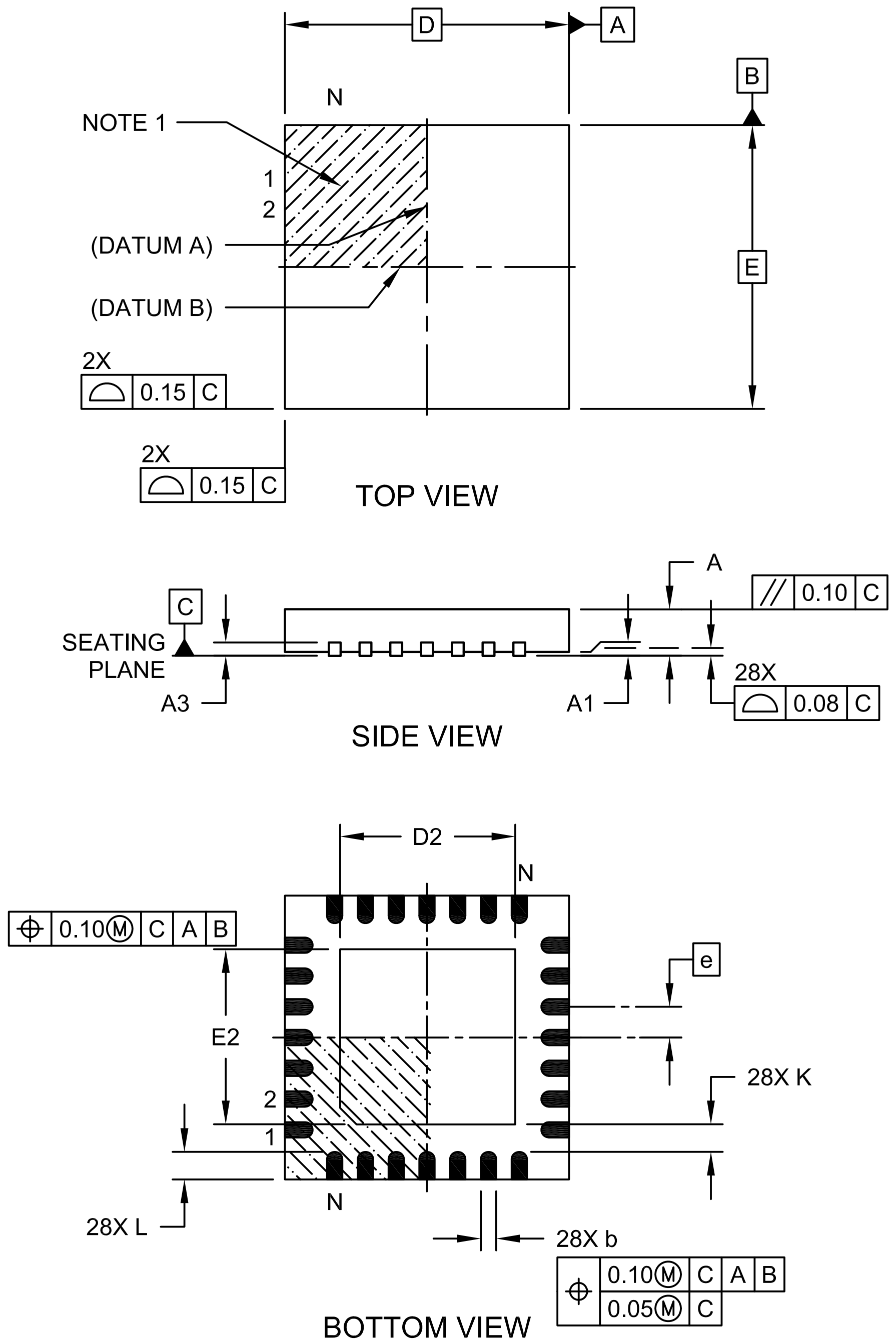
**FIGURE 37-21: $I^2C$ BUS DATA TIMING**




**Note:** Refer to Figure 37-4 for load conditions.

**28-Lead Plastic Quad Flat, No Lead Package (ML) - 6x6 mm Body [QFN] With 0.55 mm Terminal Length**

| | |
|---|---|
| **Note:** | For the most current package drawings, please see the Microchip Packaging Specification located at http://www.microchip.com/packaging |




Microchip Technology Drawing C04-105C Sheet 1 of 2