



Section 2. Oscillator

HIGHLIGHTS

This section of the manual contains the following major topics:

2.1	Introduction	2-2
2.2	Oscillator Configurations	2-2
2.3	Crystal Oscillators / Ceramic Resonators	2-4
2.4	External RC Oscillator.....	2-12
2.5	Internal 4 MHz RC Oscillator	2-13
2.6	Effects of Sleep Mode on the On-chip Oscillator	2-17
2.7	Effects of Device Reset on the On-chip Oscillator	2-17
2.8	Design Tips	2-18
2.9	Related Application Notes.....	2-19
2.10	Revision History	2-20

PICmicro MID-RANGE MCU FAMILY

2.1 Introduction

The internal oscillator circuit is used to generate the device clock. The device clock is required for the device to execute instructions and for the peripherals to function. Four device clock periods generate one internal instruction clock (Tcy) cycle.

There are up to eight different modes which the oscillator may have. There are two modes which allow the selection of the internal RC oscillator clock out (CLKOUT) to be driven on an I/O pin, or allow that I/O pin to be used for a general purpose function. The oscillator mode is selected by the device configuration bits. The device configuration bits are nonvolatile memory locations and the operating mode is determined by the value written during device programming. The oscillator modes are:

- LP Low Frequency (Power) Crystal
- XT Crystal/Resonator
- HS High Speed Crystal/Resonator
- RC External Resistor/Capacitor (same as EXTRC with CLKOUT)
- EXTRC External Resistor/Capacitor
- EXTRC External Resistor/Capacitor with CLKOUT
- INTRC Internal 4 MHz Resistor/Capacitor
- INTRC Internal 4 MHz Resistor/Capacitor with CLKOUT

These oscillator options are made available to allow a single device type the flexibility to fit applications with different oscillator requirements. The RC oscillator option saves system cost while the LP crystal option saves power. Configuration bits are used to select the various options. For more details on the device configuration bits, see the [“Device Characteristics”](#) section.

2.2 Oscillator Configurations

2.2.1 Oscillator Types

Mid-Range devices can have up to eight different oscillator modes. The user can program up to three device configuration bits (FOSC2, FOSC1 and FOSC0) to select one of these eight modes:

- LP Low Frequency (Power) Crystal
- XT Crystal/Resonator
- HS High Speed Crystal/Resonator
- RC External Resistor/Capacitor (same as EXTRC with CLKOUT)
- EXTRC External Resistor/Capacitor
- EXTRC External Resistor/Capacitor with CLKOUT
- INTRC Internal 4 MHz Resistor/Capacitor
- INTRC Internal 4 MHz Resistor/Capacitor with CLKOUT

The main difference between the LP, XT, and HS modes is the gain of the internal inverter of the oscillator circuit which allows the different frequency ranges. [Table 2-1](#) and [Table 2-2](#) give information to aid in selecting an oscillator mode. In general, use the oscillator option with the lowest possible gain which still meet specifications. This will result in lower dynamic currents (IDD). The frequency range of each oscillator mode is the recommended (tested) frequency cutoffs, but the selection of a different gain mode is acceptable as long as a thorough validation is performed (voltage, temperature, component variations (Resistor, Capacitor, and internal microcontroller oscillator circuitry)).

The RC mode and the EXTRC with CLKOUT mode have the same functionality. They are named like this to help describe their operation vs. the other oscillator modes.

Section 2. Oscillator

Table 2-1: Selecting the Oscillator Mode for Devices with FOSC1:FOSC0

Configuration bits FOSC1:FOSC0	OSC Mode	OSC Feedback Inverter Gain	Comment
1 1	RC	—	Least expensive solution for device oscillation (only an external resistor and capacitor is required). Most variation in time-base. Device's default mode.
1 0	HS	High Gain	High frequency application. Oscillator circuit's mode consumes the most current of the three crystal modes.
0 1	XT	Medium Gain	Standard crystal/resonator frequency. Oscillator circuit's mode consumes the middle current of the three crystal modes.
0 0	LP	Low Gain	Low power/frequency applications. Oscillator circuit's mode consumes the least current of the three crystal modes.

Table 2-2: Selecting the Oscillator Mode for Devices with FOSC2:FOSC0

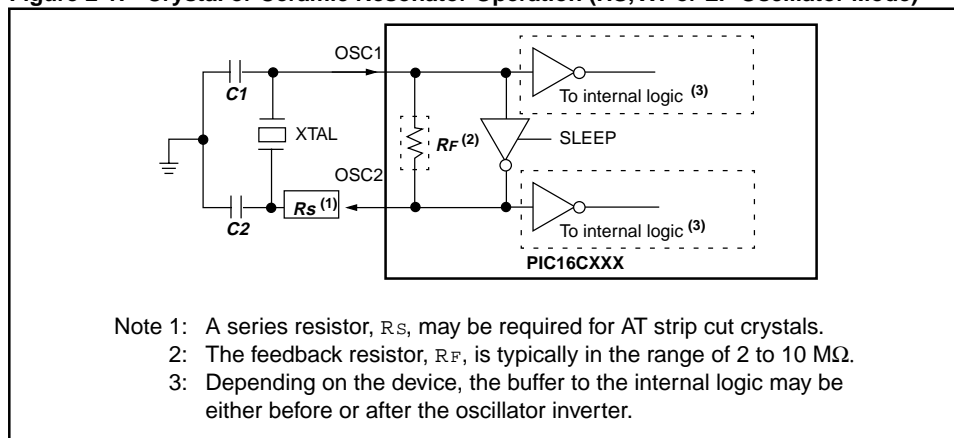
Configuration bits FOSC2:FOSC0	OSC Mode	OSC Feedback Inverter Gain	Comment
1 1 1	EXTRC with CLKOUT	—	Inexpensive solution for device oscillation. Most variation in timebase. CLKOUT is enabled on pin. Device's default mode.
1 1 0	EXTRC	—	Inexpensive solution for device oscillation. Most variation in timebase. CLKOUT is disabled (use as I/O) on pin.
1 0 1	INTRC with CLKOUT	—	Least expensive solution for device oscillation. 4 MHz oscillator, which can be tuned. CLKOUT is enabled on pin.
1 0 0	INTRC	—	Least expensive solution for device oscillation. 4 MHz oscillator, which can be tuned. CLKOUT is disabled (use as I/O) on pin.
0 1 1	—	—	Reserved
0 1 0	HS	High Gain	High frequency application. Oscillator circuit's mode consumes the most current of the three crystal modes.
0 0 1	XT	Medium Gain	Standard crystal/resonator frequency. Oscillator circuit's mode consumes the middle current of the three crystal modes.
0 0 0	LP	Low Gain	Low power/frequency applications. Oscillator circuit's mode consumes the least current of the three crystal modes.

PICmicro MID-RANGE MCU FAMILY

2.3 Crystal Oscillators / Ceramic Resonators

In XT, LP or HS modes a crystal or ceramic resonator is connected to the OSC1 and OSC2 pins to establish oscillation (Figure 2-1). The PICmicro oscillator design requires the use of a parallel cut crystal. Using a series cut crystal may give a frequency out of the crystal manufacturer's specifications. When in XT, LP or HS modes, the device can have an external clock source drive the OSC1 pin (Figure 2-3).

Figure 2-1: Crystal or Ceramic Resonator Operation (HS, XT or LP Oscillator Mode)



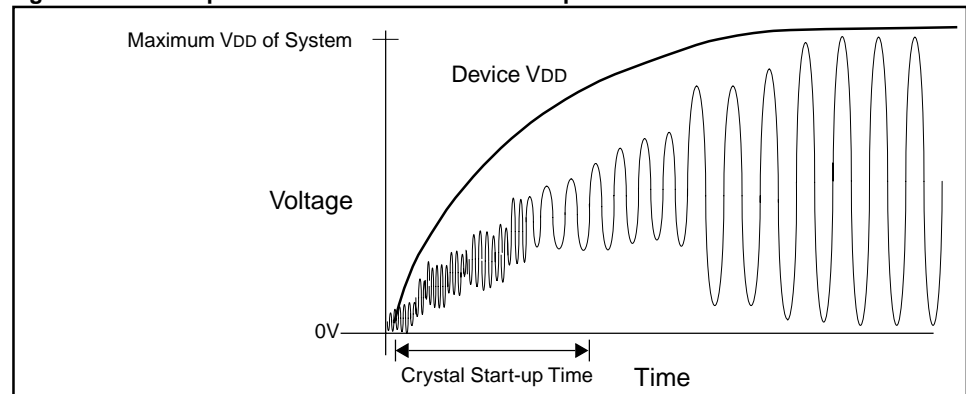
2.3.1 Oscillator / Resonator Start-up

As the device voltage increases from V_{SS} , the oscillator will start its oscillations. The time required for the oscillator to start oscillating depends on many factors. These include:

- Crystal / resonator frequency
- Capacitor values used ($C1$ and $C2$ in [Figure 2-1](#))
- Device V_{DD} rise time
- System temperature
- Series resistor value (and type) if used (R_S in [Figure 2-1](#))
- Oscillator mode selection of device (which selects the gain of the internal oscillator inverter)
- Crystal quality
- Oscillator circuit layout
- System noise

[Figure 2-2](#) graphs an example oscillator / resonator start-up. The peak-to-peak voltage of the oscillator waveform can be quite low (less than 50% of device V_{DD}) where the waveform is centered at $V_{DD}/2$ (refer to [parameters D033](#) and [D043](#) in the [“Electrical Specifications”](#) section).

Figure 2-2: Example Oscillator / Resonator Start-up Characteristics



PICmicro MID-RANGE MCU FAMILY

2.3.2 Component Selection

Figure 2-1 is a diagram of the device's crystal or ceramic resonator circuitry. The resistance for the feedback resistor, R_F , is typically within the 2 to 10 M Ω range. This varies with device voltage, temperature, and process variations. A series resistor, R_S , may be required if an AT strip cut crystal is used. Be sure to include the device's operating voltage and the device's manufacturing process when determining resistor requirements. As you can see in Figure 2-1, the connection to the device's internal logic is device dependent. See the applicable data sheet for device specifics. The typical values of capacitors (C1, C2) are given in Table 2-3 and Table 2-4. Each device's data sheet will give the specific values that Microchip tested.

Table 2-3: Typical Capacitor Selection for Ceramic Resonators

Ranges tested:		
Mode	Frequency	C1 / C2 ⁽¹⁾
XT	455 kHz	22 - 100 pF
	2.0 MHz	15 - 68 pF
	4.0 MHz	15 - 68 pF
HS	8.0 MHz	10 - 68 pF
	16.0 MHz	10 - 22 pF
	20.0 MHz	TBD
Resonators used:		
455 kHz	Panasonic EFO-A455K04B	±0.3%
2.0 MHz	Murata Erie CSA2.00MG	±0.5%
4.0 MHz	Murata Erie CSA4.00MG	±0.5%
8.0 MHz	Murata Erie CSA8.00MT	±0.5%
16.0 MHz	Murata Erie CSA16.00MX	±0.5%
20.0 MHz	TBD	TBD

Note 1: Recommended values of C1 and C2 are identical to the ranges tested above.

Higher capacitance increases the stability of the oscillator but also increases the start-up time. These values are for design guidance only. Since each resonator has its own characteristics, the user should consult the resonator manufacturer for appropriate values of external component or verify oscillator performance.

2: All resonators tested required external capacitors.

Section 2. Oscillator

Table 2-4: Typical Capacitor Selection for Crystal Oscillator

Mode	Freq	C1 ⁽¹⁾	C2 ⁽¹⁾
LP	32 kHz	68 - 100 pF	68 - 100 pF
	200 kHz	15 - 30 pF	15 - 30 pF
XT	100 kHz	68 - 150 pF	150 - 200 pF
	2 MHz	15 - 30 pF	15 - 30 pF
	4 MHz	15 - 30 pF	15 - 30 pF
HS	8 MHz	15 - 30 pF	15 - 30 pF
	10 MHz	15 - 30 pF	15 - 30 pF
	20 MHz	15 - 30 pF	15 - 30 pF
Crystals used:			
32.768 kHz	Epson C-001R32.768K-A		± 20 PPM
100 kHz	Epson C-2 100.00 KC-P		± 20 PPM
200 kHz	STD XTL 200.000 kHz		± 20 PPM
2.0 MHz	ECS ECS-20-S-2		± 50 PPM
4.0 MHz	ECS ECS-40-S-4		± 50 PPM
10.0 MHz	ECS ECS-100-S-4		± 50 PPM
20.0 MHz	ECS ECS-200-S-4		± 50 PPM

Note 1: Higher capacitance increases the stability of the oscillator but also increases the start-up time. These values are for design guidance only. A series resistor, R_S , may be required in HS mode as well as XT mode to avoid overdriving crystals with low drive level specification. Since each crystal has its own characteristics, the user should consult the crystal manufacturer for appropriate values of external components or verify oscillator performance.

2.3.3 Tuning the Oscillator Circuit

Since Microchip devices have wide operating ranges (frequency, voltage, and temperature; depending on the part and version ordered) and external components (crystals, capacitors,...), of varying quality and manufacture; validation of operation needs to be performed to ensure that the component selection will comply with the requirements of the application.

There are many factors that go into the selection and arrangement of these external components. These factors include:

- amplifier gain
- desired frequency
- resonant frequency(s) of the crystal
- temperature of operation
- supply voltage range
- start-up time
- stability
- crystal life
- power consumption
- simplification of the circuit
- use of standard components
- combination which results in fewest components

2.3.3.1 Determining Best Values for Crystals, Clock Mode, C_1 , C_2 , and R_S

The best method for selecting components is to apply a little knowledge and a lot of trial, measurement, and testing.

Crystals are usually selected by their parallel resonant frequency only, however other parameters may be important to your design, such as temperature or frequency tolerance. Application Note AN588 is an excellent reference if you would like to know more about crystal operation and their ordering information.

The PICmicroTM internal oscillator circuit is a parallel oscillator circuit, which requires that a parallel resonant crystal be selected. The load capacitance is usually specified in the 20 pF to 32 pF range. The crystal will oscillate closest to the desired frequency with capacitance in this range. It may be necessary to sometimes juggle these values a bit, as described later, in order to achieve other benefits.

Clock mode is primarily chosen by using the FOSC parameter specification ([parameter 1A](#)) in the device's data sheet, based on frequency. Clock modes (except RC) are simply gain selections, lower gain for lower frequencies, higher gain for higher frequencies. It is possible to select a higher or lower gain, if desired, based on the specific needs of the oscillator circuit.

C_1 and C_2 should also be initially selected based on the load capacitance as suggested by the crystal manufacturer and the tables supplied in the device data sheet. The values given in the Microchip data sheet can only be used as a starting point, since the crystal manufacturer, supply voltage, and other factors already mentioned may cause your circuit to differ from the one used in the factory characterization process.

Ideally, the capacitance is chosen (within the range of the recommended crystal load preferably) so that it will oscillate at the highest temperature and lowest V_{DD} that the circuit will be expected to perform under. High temperature and low V_{DD} both have a limiting affect on the loop gain, such that if the circuit functions at these extremes, the designer can be more assured of proper operation at other temperatures and supply voltage combinations. The output sine wave should not be clipped in the highest gain environment (highest V_{DD} and lowest temperature) and the sine output amplitude should be great enough in the lowest gain environment (lowest V_{DD} and highest temperature) to cover the logic input requirements of the clock as listed in the device data sheet.

A method for improving start-up is to use a value of C_2 greater than C_1 . This causes a greater phase shift across the crystal at power-up, which speeds oscillator start-up.

Besides loading the crystal for proper frequency response, these capacitors can have the effect of lowering loop gain if their value is increased. C_2 can be selected to affect the overall gain of the circuit. A higher C_2 can lower the gain if the crystal is being over driven (see also discussion on R_S). Capacitance values that are too high can store and dump too much current through the crystal, so C_1 and C_2 should not become excessively large. Unfortunately, measuring the wattage through a crystal is tricky business, but if you do not stray too far from the suggested values you should not have to be concerned with this.

A series resistor, R_S , is added to the circuit if, after all other external components are selected to satisfaction, the crystal is still being over driven. This can be determined by looking at the OSC2 pin, which is the driven pin, with an oscilloscope. Connecting the probe to the OSC1 pin will load the pin too much and negatively affect performance. Remember that a scope probe adds its own capacitance to the circuit, so this may have to be accounted for in your design, i.e. if the circuit worked best with a C_2 of 20 pF and scope probe was 10 pF, a 30 pF capacitor may actually be called for. The output signal should not be clipping or squashed. Overdriving the crystal can also lead to the circuit jumping to a higher harmonic level or even crystal damage.

PICmicro MID-RANGE MCU FAMILY

The OSC2 signal should be a nice clean sine wave that easily spans the input minimum and maximum of the clock input pin (4V to 5V peak to peak for a 5V V_{DD} is usually good). An easy way to set this is to again test the circuit at the minimum temperature and maximum V_{DD} that the design will be expected to perform in, then look at the output. This should be the maximum amplitude of the clock output. If there is clipping or the sine wave is squashing near V_{DD} and V_{SS} at the top and bottom, and increasing load capacitors will risk too much current through the crystal or push the value too far from the manufacturer's load specification, then add a trimpot between the output pin and $C2$, and adjust it until the sine wave is clean. Keeping it fairly close to maximum amplitude at the low temperature and high V_{DD} combination will assure this is the maximum amplitude the crystal will see and prevent overdriving. A series resistor, R_S , of the closest standard value, can now be inserted in place of the trimpot. If R_S is too high, perhaps more than 20k ohms, the input will be too isolated from the output, making the clock more susceptible to noise. If you find a value this high is needed to prevent overdriving the crystal, try increasing $C2$ to compensate. Try to get a combination where R_S is around 10k or less, and load capacitance is not too far from the 20 pF or 32 pF manufacturer specification.

2.3.3.1.1 Start-up

The most difficult time for the oscillator to start-up is when waking up from sleep. This is because the load capacitors have both partially charged to some quiescent value, and phase differential at wake-up is minimal. Thus, more time is required to achieve stable oscillation. Remember also that low voltage, high temperatures, and the lower frequency clock modes also impose limitations on loop gain, which in turn affects start-up. Each of the following factors makes thing worse:

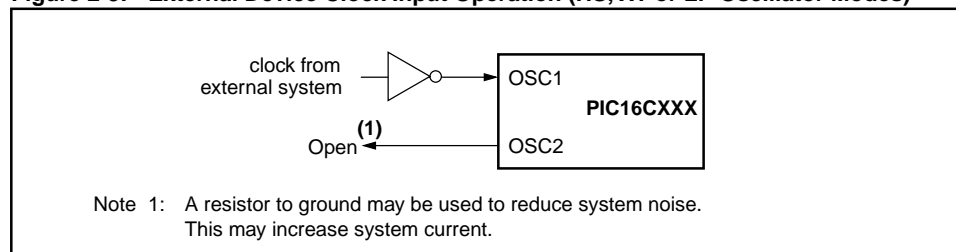
- a low frequency design (with its low gain clock mode)
- a quiet environment (such as a battery operated device)
- operating outside the noisy RF area (such as in a shielded box)
- low voltage
- high temperature
- waking up from sleep.

Noise actually helps a design for oscillator start-up, since it helps kick start the oscillator.

2.3.4 External Clock Input

If the PICmicro's internal oscillator is not being used, and the device will be driven from an external clock, be sure to set the oscillator mode to one of the crystal modes (LP, XT, or HS). That is, something other than one of the RC modes, since RC mode will fight with the injected input. Ideally you would select the mode that corresponds to the frequency injected, but this is of less importance here since the clock is only driving its internal logic, and not a crystal loop circuit. It may be possible to select a clock mode lower than would be needed by an oscillator circuit, and thereby save some of the power that would be used exercising the inverting amplifier. Make sure the OSC2 signal amplitude covers the needed logic thresholds of the device.

Figure 2-3: External Device Clock Input Operation (HS, XT or LP Oscillator Modes)



2.3.5 External Crystal Oscillator Circuit for Device Clock

Sometimes more than one device needs to be clocked from a single crystal. Since Microchip does not recommend connecting other logic to the PICmicro's internal oscillator circuit, an external crystal oscillator circuit is recommended. Each device will then have an external clock source, and the number of devices that can be driven will depend on the buffer drive capability. This circuit is also useful when more than one device (PICmicro) needs to operate synchronously to each other.

Either a prepackaged oscillator can be used or a simple oscillator circuit with TTL gates can be built. Prepackaged oscillators provide a wide operating range and better stability. A well-designed crystal oscillator will provide good performance with TTL gates. Two types of crystal oscillator circuits can be used; one with series resonance, or one with parallel resonance.

Figure 2-4 shows implementation of an external parallel resonant oscillator circuit. The circuit is designed to use the fundamental frequency of the crystal. The 74AS04 inverter performs the 180-degree phase shift that a parallel oscillator requires. The 4.7 k Ω resistor provides the negative feedback for stability. The 10 k Ω potentiometer biases the 74AS04 in the linear region.

Figure 2-4: External Parallel Resonant Crystal Oscillator Circuit

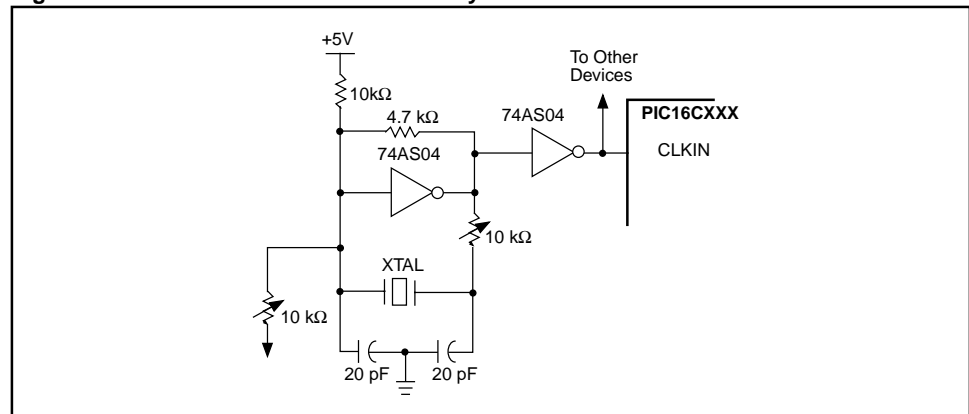
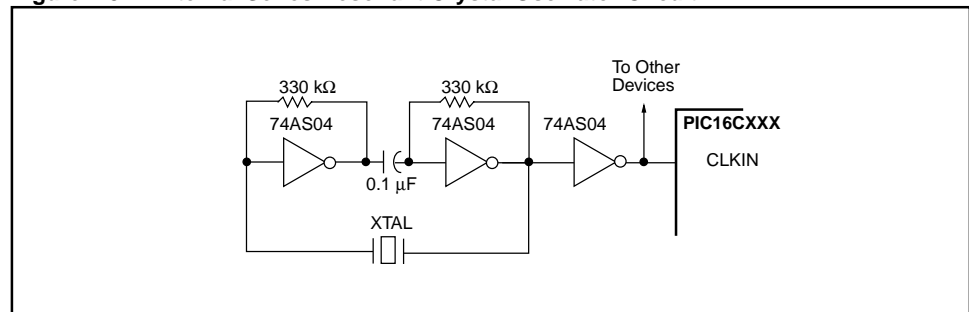


Figure 2-5 shows an external series resonant oscillator circuit. This circuit is also designed to use the fundamental frequency of the crystal. The inverter performs a 180-degree phase shift in a series resonant oscillator circuit. The 330 k Ω resistors provide the negative feedback to bias the inverters in their linear region.

Figure 2-5: External Series Resonant Crystal Oscillator Circuit



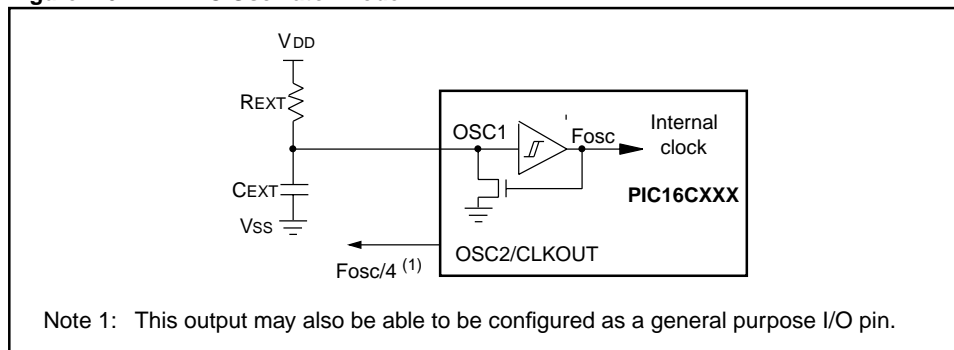
When the device is clocked from an external clock source (as in Figure 2-4 or Figure 2-5) then the microcontroller's oscillator must be configured for LP, XT or HS mode (Figure 2-3).

PICmicro MID-RANGE MCU FAMILY

2.4 External RC Oscillator

For timing insensitive applications the “EXTRC” device option offers additional cost savings. The RC oscillator frequency is a function of; the supply voltage, the resistor (R_{EXT}) and capacitor (C_{EXT}) values, and the operating temperature. In addition to this, the oscillator frequency will vary from unit to unit due to normal process parameter variation. Furthermore, the difference in lead frame capacitance between package types will also affect the oscillation frequency, especially for low C_{EXT} values. The user also needs to take into account variation due to tolerance of external R_{EXT} and C_{EXT} components used. Figure 2-6 shows how the RC combination is connected to a PIC16CXXX. For R_{EXT} values below 2.2 k Ω , oscillator operation may become unstable, or stop completely. For very high R_{EXT} values (e.g. 1 M Ω), the oscillator becomes sensitive to noise, humidity and leakage. Thus, we recommend keeping R_{EXT} between 3 k Ω and 100 k Ω .

Figure 2-6: EXTRC Oscillator Mode



Although the oscillator will operate with no external capacitor (C_{EXT} = 0 pF), we recommend using values above 20 pF for noise and stability reasons. With no or small external capacitance, the oscillation frequency can vary dramatically due to changes in external capacitances, such as PCB trace capacitance and package lead frame capacitance.

See characterization data for RC frequency variation from part to part due to normal process variation. The variation is larger for larger resistance (since leakage current variation will affect RC frequency more for large R) and for smaller capacitance (since variation of input capacitance will affect RC frequency more).

See characterization data for variation of oscillator frequency due to V_{DD} for given R_{EXT}/C_{EXT} values as well as frequency variation due to operating temperature for given R_{EXT}, C_{EXT}, and V_{DD} values.

The oscillator frequency, divided by 4, is available on the OSC2/CLKOUT pin, and can be used for test purposes or to synchronize other logic (see Figure 4-3: “Clock/Instruction Cycle” in the “Architecture” section, for waveform).

2.4.1 RC Start-up

As the device voltage increases, the RC will start its oscillations immediately after the pin voltage levels meet the input threshold specifications (parameters D032 and D042 in the “Electrical Specifications” section). The time required for the RC to start oscillating depends on many factors. These include:

- Resistor value used
- Capacitor value used
- Device V_{DD} rise time
- System temperature

2.5 Internal 4 MHz RC Oscillator

The internal RC oscillator (not on all devices) provides a fixed 4 MHz (nominal) system clock at $V_{DD} = 5V$ and $25^{\circ}C$, see the device data sheet's "Electrical Specifications" section for information on variation over voltage and temperature.

The value in the OSCCAL register is used to tune the frequency of the internal RC oscillator. The calibration value that Microchip programs into the device will "trim" the internal oscillator to remove process variation from the oscillator frequency. The CAL3:CAL0 bits are used for fine calibration within a frequency window. Higher values of CAL3:CAL0 (from 0000 to 1111) yields higher clock speeds.

When a 4 MHz internal RC oscillator frequency cannot be achieved by a CAL3:CAL0 value, the RC oscillator frequency can be increased or decreased by an offset frequency. The CALFST and CALSLW bits are used to enable a positive or negative frequency offset to place the internal RC frequency within the CAL3:CAL0 trim window.

Setting the CALFST bit offsets the internal RC for a higher frequency, while setting the CALSLW bit offsets the internal RC for a lower frequency.

Upon a device reset, the OSCCAL register is forced to the midpoint value (CAL3:CAL0 = 7h, CALFST and CALSLW providing no offset).

Register 2-1: OSCCAL Register

R/W-0	R/W-1	R/W-1	R/W-1	R/W-0	R/W-0	U-0	U-0
CAL3	CAL2	CAL1	CAL0	CALFST	CALSLW	—	—
bit 7						bit 0	

bit 7:4 **CAL3:CAL0:** Internal RC Oscillator Calibration bits
0000 = Lowest clock frequency within the trim range

•
•
•

1111 = Highest clock frequency within the trim range

bit 3 **CALFST:** Oscillator Range Offset bit
1 = Increases the frequency of the internal RC oscillator into the CAL3:CAL0 trim window
0 = No offset provided

bit 2 **CALSLW:** Oscillator Range Offset bit
1 = Decreases the frequency of the internal RC oscillator into the CAL3:CAL0 trim window
0 = No offset provided

Note: When both bits are set, the CALFST bit overrides the CALSLW bit.

bit 1:0 **Unimplemented:** Read as '0'

Note: These bits should be written as '0' when modifying the OSCCAL register, for compatibility with future devices.

Legend

R = Readable bit

W = Writable bit

U = Unimplemented bit, read as '0'

- n = Value at POR reset

Note: OSCCAL is used to remove process variation from the internal RC oscillator of the device. The OSCCAL value should not be modified from the Microchip supplied value, and all timing critical functions should be adjusted by the application software.

PICmicro MID-RANGE MCU FAMILY

Figure 2-7 shows the possible device frequencies from the uncalibrated point (at $V_{DD} = 5V$, $25^{\circ}C$, and $OSCCAL = 70h$), and the changes achievable by the $OSCCAL$ register.

Figure 2-7: Ideal Internal RC Oscillator Frequency vs. $OSCCAL$ Register Value

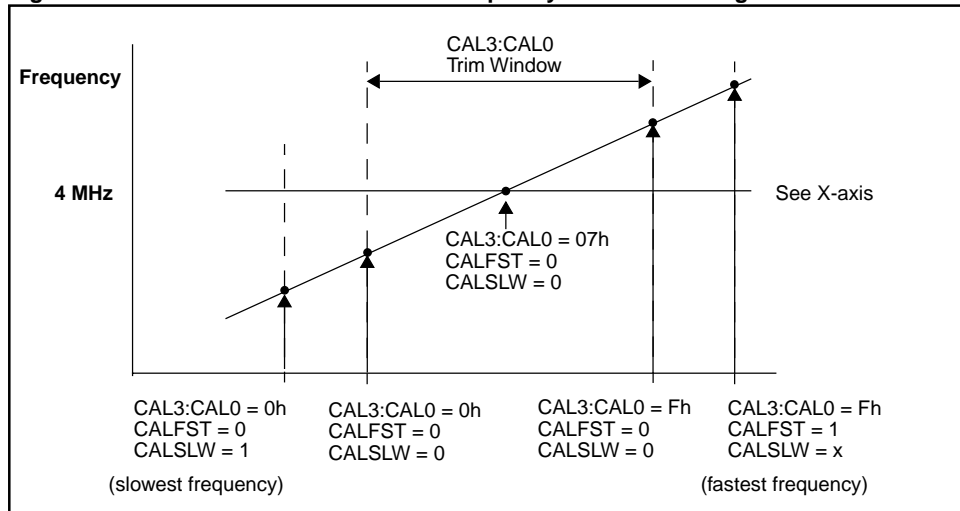


Figure 2-8 shows an example of a device where by selecting one of the $CAL3:CAL0$ values, the frequency can be corrected to 4 MHz. These bits can be considered the fine trimming of the frequency. Sometimes the device frequency at the uncalibrated point cannot be corrected to 4 MHz by the fine trimming of the $CAL3:CAL0$ bits value. Therefore two additional bits are available which give a large frequency offset (positive and negative) to move the frequency within the range where the fine trimming can work. These bits are the $CALSLW$ and $CALFST$ bits, which offset the internal RC frequency. The action of these bits are shown in Figure 2-9, and Figure 2-10.

Figure 2-8: $CAL3:CAL0$ Trimming of Internal RC Oscillator Frequency Offset

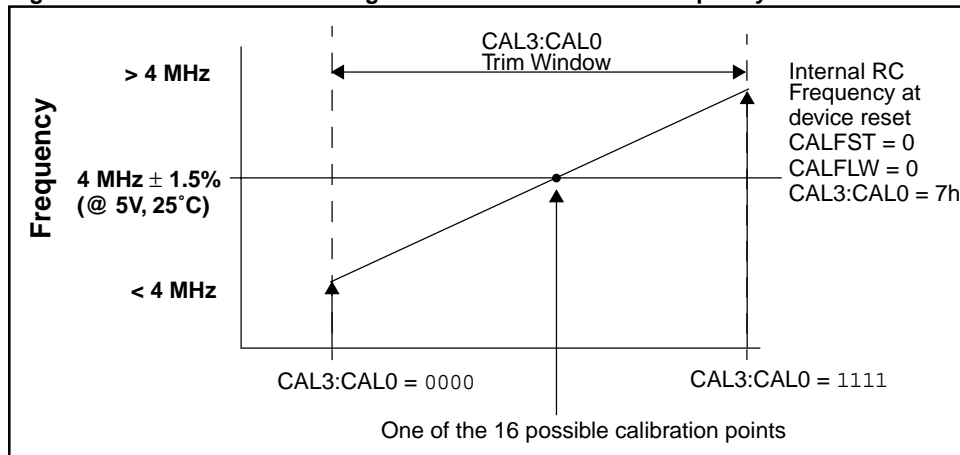


Figure 2-9: CALFST Positive Internal RC Oscillator Frequency Offset

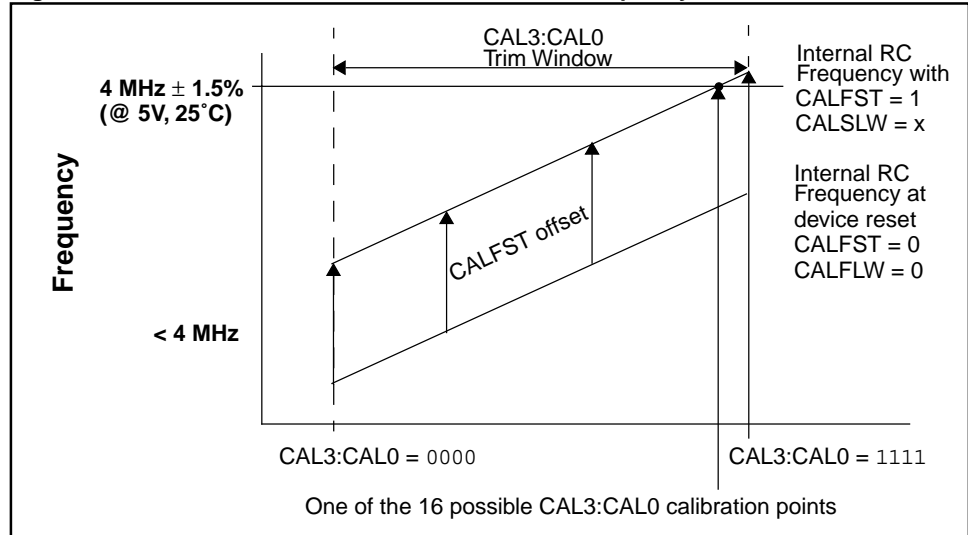
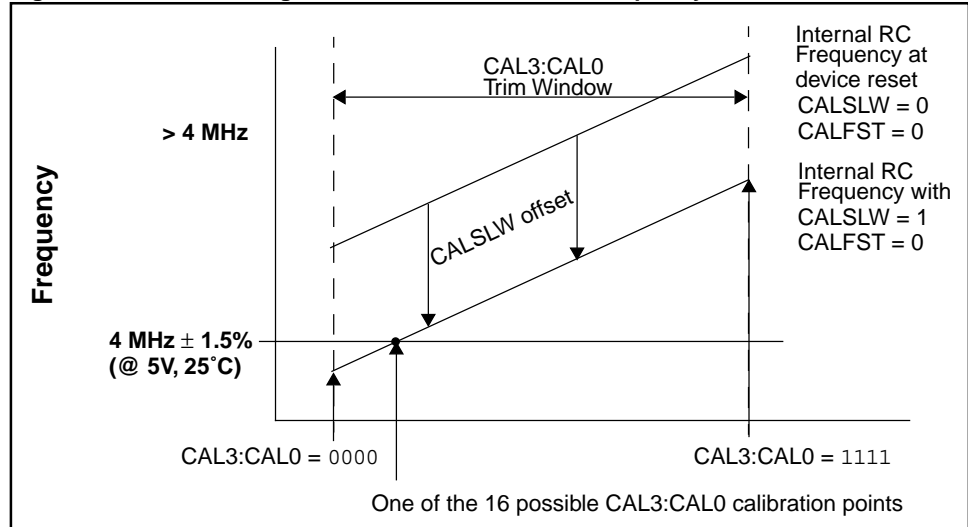


Figure 2-10: CALSLW Negative Internal RC Oscillator Frequency Offset



PICmicro MID-RANGE MCU FAMILY

A calibration instruction is programmed into the last address of the implemented program memory. This instruction contains the calibration value for the internal RC oscillator. This value is programmed as a `RETLW XX` instruction where `XX` is the calibration value. In order to retrieve the calibration value, issue a `CALL YY` instruction where `YY` is the last location in the device's user accessible program memory. The calibration value is now loaded in the `W` register. The program should then perform a `MOVWF OSCCAL` instruction to load the value into the internal RC oscillator calibration register. Table 2-5 shows the location of the calibration value depending on the size of the program memory.

Table 2-5: Calibration Value Location

Program Memory Size (Words)	Calibration Value Location
512	1FFh
1K	3FFh
2K	7FFh
4K	FFFh
8K	1FFFh

Note 1: Erasing the device (windowed devices) will also erase the factory programmed calibration value for the internal oscillator.

Prior to erasing a windowed device, the internal oscillator calibration value must be saved. It is a good idea to write this value on the package of the device to ensure that the calibration value is not accidentally lost.

This saved value must be restored into program memory calibration location before programming the device.

Note 2: `OSCCAL<1:0>` are unimplemented and should be written as '0'. This will help ensure compatibility with future devices.

2.5.1 Clock Out

The internal RC oscillator can be configured to provide a clock out signal on the `CLKOUT` pin when the configuration word address (2007h) is programmed with `FOSC2`, `FOSC1`, `FOSC0` equal to '101' for Internal RC or '111' for External RC. `CLKOUT`, which is divided by 4, can be used for test purposes or to synchronize other logic.

When the calibration value of the internal RC oscillator is accidentally erased, the clock out feature allows the user to determine what the calibration value should be. This is achieved by writing a program which modifies (increments/decrements) the value of the `OSCCAL` register. When the `CLKOUT` pin is at 4 MHz ($\pm 1.5\%$) at 5V and 25°C, the `OSCCAL` register has the correct calibration value. This value then needs to be written to a port or shifted out serially, so that the value can be written down and programmed into the calibration location.

2.6 Effects of Sleep Mode on the On-chip Oscillator

When the device executes a **SLEEP** instruction, the on-chip clocks and oscillator are turned off and the device is held at the beginning of an instruction cycle (Q1 state). With the oscillator off, the OSC1 and OSC2 signals will stop oscillating. Since all the transistor switching currents have been removed, sleep mode achieves the lowest current consumption of the device (only leakage currents). Enabling any on-chip feature that will operate during sleep will increase the current consumed during sleep. The user can wake from SLEEP through external reset, Watchdog Timer Reset or through an interrupt.

Table 2-6: OSC1 and OSC2 Pin States in Sleep Mode

OSC Mode	OSC1 Pin	OSC2 Pin
EXTRC	Floating, external resistor should pull high	At logic low
INTRC	N.A.	N.A.
LP, XT, and HS	Feedback inverter disabled, at quiescent voltage level	Feedback inverter disabled, at quiescent voltage level

See [Table 3-1](#), in the “**Reset**” section, for time-outs due to Sleep and MCLR reset.

2.7 Effects of Device Reset on the On-chip Oscillator

Device resets have no effect on the on-chip crystal oscillator circuitry. The oscillator will continue to operate as it does under normal execution. While in reset, the device logic is held at the Q1 state so that when the device exits reset, it is at the beginning of an instruction cycle.

The OSC2 pin, when used as the external clockout (EXTRC mode), will be held low during reset, and as soon as the $\overline{\text{MCLR}}$ pin is at V_{IH} (input high voltage), the RC will start to oscillate.

See [Table 3-1](#), in the “**Reset**” section, for time-outs due to Sleep and $\overline{\text{MCLR}}$ reset.

2.7.1 Power-up Delays

There are two timers that offer necessary delays on power-up. One is the Oscillator Start-up Timer, OST, intended to keep the chip in RESET until the crystal oscillator is stable. The other is the Power-up Timer (PWRT), which provides a fixed delay of 72 ms (nominal) on power-up only (POR and BOR). The PWRT is designed to keep the part in RESET while the power supply stabilizes. With these two timers on-chip, most applications need no external reset circuitry. For additional information on reset operation, see the “**Reset**” section.

2.8 Design Tips

Question 1: *When looking at the OSC2 pin after power-up with an oscilloscope, there is no clock. What can cause this?*

Answer 1:

1. Executing a `SLEEP` instruction with no source for wake-up (such as, `WDT`, `MCLR`, or an Interrupt). Verify that the code does not put device to sleep without providing for wake-up. If it is possible, try waking it up with a low pulse on `MCLR`. Powering up with `MCLR` held low will also give the crystal oscillator more time to start-up, but the Program Counter will not advance until the `MCLR` pin is high.
2. The wrong clock mode is selected for the desired frequency. For a blank device, the default oscillator is `EXTRC`. Most parts come with the clock selected in the default `RC` mode, which will not start oscillation with a crystal or resonator. Verify that the clock mode has been programmed correctly.
3. The proper power-up sequence has not been followed. If a CMOS part is powered through an I/O pin prior to power-up, bad things can happen (latch up, improper start-up etc.) It is also possible for brown-out conditions, noisy power lines at start-up, and slow `VDD` rise times to cause problems. Try powering up the device with nothing connected to the I/O, and power-up with a known, good, fast-rise, power supply. It is not as much of a problem as it may sound, but the possibility exists. Refer to the power-up information in the device data sheet for considerations on brown-out and power-up sequences.
4. The `C1` and `C2` capacitors attached to the crystal have not been connected properly or are not the correct values. Make sure all connections are correct. The device data sheet values for these components will almost always get the oscillator running, they just might not be the optimal values for your design.

Question 2: *The PICmicro starts, but runs at a frequency much higher than the resonant frequency of the crystal.*

Answer 2:

The gain is too high for this oscillator circuit. Refer to subsection [2.3 “Crystal Oscillators / Ceramic Resonators”](#) to aid in the selection of `C2` (may need to be higher) `RS` (may be needed) and clock mode (wrong mode may be selected). This is especially possible for low frequency crystals, like the common 32.768 kHz.

Question 3: *The design runs fine, but the frequency is slightly off, what can be done to adjust this?*

Answer 3:

Changing the value of `C1` has some affect on the oscillator frequency. If a `SERIES` resonant crystal is used, it will resonate at a different frequency than a `PARALLEL` resonant crystal of the same frequency call-out.

Question 4: *The board works fine, then suddenly quits, or loses time.*

Answer 4:

Other than the obvious software checks that should be done to investigate losing time, it is possible that the amplitude of the oscillator output is not high enough to reliably trigger the oscillator input.

Question 5: *I'm using a device with the internal RC oscillator and I have accidentally erased the calibration value. What can I do?*

Answer 5:

If the frequency of the device does not matter, you can continue to use the device.

If the frequency of the device does matter, you can purchase a new windowed device, or follow the suggestion in subsection [2.5.1 “Clock Out.”](#)

2.9 Related Application Notes

This section lists application notes that are related to this section of the manual. These application notes may not be written specifically for the Mid-Range MCU family (that is they may be written for the Base-Line, or High-End families), but the concepts are pertinent, and could be used (with modification and possible limitations). The current application notes related to the oscillator are:

Title	Application Note #
PIC16/17 Oscillator Design Guide	AN588
Low Power Design using PIC16/17	AN606

PICmicro MID-RANGE MCU FAMILY

2.10 Revision History

Revision A

This is the initial released revision of the PICmicro oscillators description.