# rfPIC12C509AG/509AF

# Data Sheet

## 18/20-Pin 8-Bit CMOS Microcontroller with UHF ASK/FSK Transmitter

**Note the following details of the code protection feature on PICmicro® MCUs.**

- The PICmicro family meets the specifications contained in the Microchip Data Sheet.
- Microchip believes that its family of PICmicro microcontrollers is one of the most secure products of its kind on the market today, when used in the intended manner and under normal conditions.
- There are dishonest and possibly illegal methods used to breach the code protection feature. All of these methods, to our knowledge, require using the PICmicro microcontroller in a manner outside the operating specifications contained in the data sheet. The person doing so may be engaged in theft of intellectual property.
- Microchip is willing to work with the customer who is concerned about the integrity of their code.
- Neither Microchip nor any other semiconductor manufacturer can guarantee the security of their code. Code protection does not mean that we are guaranteeing the product as “unbreakable”.
- Code protection is constantly evolving. We at Microchip are committed to continuously improving the code protection features of our product.

If you have any further questions about this matter, please contact the local sales office nearest to you.



DNV Certification, Inc. USA — ANSI-RAB QMS ACCREDITED — DNV MSC The Netherlands Accredited by the RvA — DNV — ISO 9001 / QS-9000 REGISTERED FIRM

*Microchip received QS-9000 quality system certification for its worldwide headquarters, design and wafer fabrication facilities in Chandler and Tempe, Arizona in July 1999. The Company’s quality system processes and procedures are QS-9000 compliant for its PICmicro® 8-bit MCUs, KEELOQ® code hopping devices, Serial EEPROMs and microperipheral products. In addition, Microchip’s quality system for the design and manufacture of development systems is ISO 9001 certified.*

# rfPIC12C509AG/509AF

## 8-Bit CMOS Microcontroller with UHF ASK/FSK Transmitter

### High-Performance RISC CPU:

- Only 33 single word instructions to learn
- All instructions are single cycle (1 µs) except for program branches which are two-cycle
- Operating speed: DC - 4 MHz clock input
  DC - 1 µs instruction cycle

| Device | Memory | | |
|---|---|---|---|
| | EPROM Program | RAM Data | Transmitter |
| rfPIC12C509AG | 1024 x 12 | 41 | ASK |
| rfPIC12C509AF | 1024 x 12 | 41 | ASK/FSK |

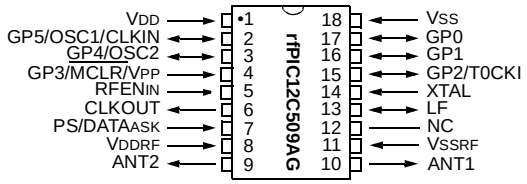
- 12-bit wide instructions
- 8-bit wide data path
- Seven special function hardware registers
- Two-level deep hardware stack
- Direct, indirect and relative addressing modes for data and instructions
- Internal 4 MHz RC oscillator with programmable calibration (independent from transmitter quartz crystal reference)
- In-Circuit Serial Programming™ (ICSP™)

### Peripheral Features:

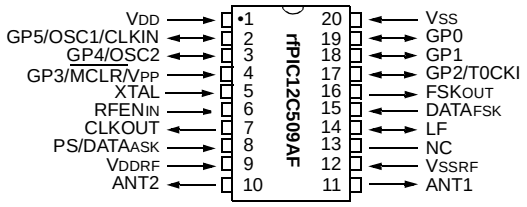
- 8-bit real time clock/counter (TMR0) with 8-bit programmable prescaler
- Power-On Reset (POR)
- Device Reset Timer (DRT)
- Watchdog Timer (WDT) with its own on-chip RC oscillator for reliable operation
- Programmable code-protection
- Power saving SLEEP mode
- Wake-up from SLEEP on pin change
- Internal weak pull-ups on I/O pins
- Internal pull-up on $\overline{MCLR}$ pin
- Selectable oscillator options:
  - INTRC: Internal 4 MHz RC oscillator
  - EXTRC: External low-cost RC oscillator
  - XT: Standard crystal/resonator
  - LP: Power saving, low frequency crystal

### Pin Diagram

**SOIC**




**SSOP**




### UHF ASK/FSK Transmitter:

- Conforms to US FCC Part 15.231 regulations and European ERC 70-03E and EN 300 220-1 requirements
- VCO phase locked to quartz crystal reference; allows narrow band receivers to be used to maximize range and interference immunity
- Integrated crystal oscillator and VCO requiring minimum of external components
- Crystal frequency divide by 4 available (CLKOUT)
- Frequency range set by crystal: 310 – 480 MHz
- ASK Data rate: 0 – 40 Kbps
- FSK through crystal pulling allows modulation at 0 – 20 Kbps
- Adjustable output power: +2 dBm to -12 dBm in six discrete steps
- Differential output configurable for single or double ended loop antenna
- Power amplifier automatically disabled until after PLL lock

### CMOS Technology:

- Low power, high speed CMOS EPROM technology
- Fully static design
- Wide operating voltage range
- Wide temperature range:
  - Industrial: -40°C to +85°C
- PICmicro® MCU power consumption:
  - < 2 mA @ 5V, 4 MHz
  - 15 μA typical @ 3V, 32 KHz
  - < 1 μA typical standby current
- Transmitter power consumption: (depending on power selection)
  - 4.8 mA to 11.5 mA @ 3V
  - <1 μA typical standby current

## TABLE OF CONTENTS



## TO OUR VALUED CUSTOMERS

It is our intention to provide our valued customers with the best documentation possible to ensure successful use of your Microchip products. To this end, we will continue to improve our publications to better suit your needs. Our publications will be refined and enhanced as new volumes and updates are introduced.

If you have any questions or comments regarding this publication, please contact the Marketing Communications Department via E-mail at **docerrors@mail.microchip.com** or fax the **Reader Response Form** in the back of this data sheet to (480) 792-4150. We welcome your feedback.

### Most Current Data Sheet

To obtain the most up-to-date version of this data sheet, please register at our Worldwide Web site at:

http://www.microchip.com

You can determine the version of a data sheet by examining its literature number found on the bottom outside corner of any page. The last character of the literature number is the version number, (e.g., DS30000A is version A of document DS30000).

### Errata

An errata sheet, describing minor operational differences from the data sheet and recommended workarounds, may exist for current devices. As device/documentation issues become known to us, we will publish an errata sheet. The errata will specify the revision of silicon and revision of document to which it applies.

To determine if an errata sheet exists for a particular device, please check with one of the following:

- Microchip's Worldwide Web site; http://www.microchip.com
- Your local Microchip sales office (see last page)
- The Microchip Corporate Literature Center; U.S. FAX: (480) 792-7277

When contacting a sales office or the literature center, please specify which device, revision of silicon and data sheet (include literature number) you are using.

### Customer Notification System

Register on our web site at **www.microchip.com/cn** to receive the most current information on all of our products.

## 1.0 GENERAL DESCRIPTION

The rfPIC12C509AG/509AF from Microchip Technology is a low-cost, high performance, 8-bit, fully static, EPROM-based CMOS microcontroller combined with a UHF ASK/FSK transmitter. It employs a RISC architecture with only 33 single word/single cycle instructions. All instructions are single cycle (1 µs) except for program branches which take two cycles. The 12-bit wide instructions are highly symmetrical resulting in 2:1 code compression over other 8-bit microcontrollers in its class. The easy to use and easy to remember instruction set reduces development time significantly.

The rfPIC12C509AG/509AF product is equipped with special features that reduce system cost and power requirements. The Power-on Reset (POR) and Device Reset Timer (DRT) eliminate the need for external RESET circuitry. There are four oscillator configurations to choose from, including INTRC internal oscillator mode and the power-saving LP (Low Power) oscillator mode. Power saving SLEEP mode, Watchdog Timer and code protection features also improve system cost, power and reliability.

The Transmitter is a fully integrated UHF ASK/FSK transmitter consisting of crystal oscillator, phase-locked loop (PLL), open-collector differential-output Power Amplifier (PA), and mode control logic. External components consist of bypass capacitors, crystal, and PLL loop filter. There are no internal electrical connections between the PICmicro MCU and the transmitter. The PICmicro MCU oscillator is independent from the transmitter crystal oscillator.

The rfPIC12C509AG is capable of Amplitude Shift Keying (ASK) modulation by turning the PA on and off. The rfPIC12C509AF is capable of ASK or Frequency Shift Keying (FSK) modulation by employing an internal FSK switch to pull the transmitter crystal via a second load capacitor.

The rfPIC12C509AG/509AF is a single channel device. The transmit frequency is fixed and set by an external reference crystal. Transmit frequencies in the range of 310 to 480 MHz can be selected. Output drive is an open-collector differential amplifier. The differential output is well suited for loop antennas. Output power is adjustable from +2 dBm to -12 dBm in six discrete steps.

The rfPIC12C509AG/509AF are radio frequency (RF) emitting devices. Wireless RF devices are governed by a country's regulating agency. For example, in the United States it is the Federal Communications Committee (FCC) and in Europe it is the European Conference of Postal and Telecommunications Administrations (CEPT). It is the responsibility of the designer to ensure that their end product conforms to rules and regulations of the country of use and/or sale. RF devices require correct board level implementation in order to meet regulatory requirements. Layout considerations are given in Section 7.0 UHF ASK/FSK Transmitter.

The rfPIC12C509AG/509AF is available in the cost-effective One-Time-Programmable (OTP) version which is suitable for production in any volume. The customer can take full advantage of Microchip's price leadership in OTP microcontrollers while benefiting from the OTP's flexibility.

The rfPIC12C509AG/509AF product is supported by a full-featured macro assembler, a software simulator, an in-circuit emulator, a 'C' compiler, a low-cost development programmer, and a full featured programmer. All the tools are supported on IBM® PC and compatible machines.

### 1.1 Applications

The rfPIC12C509AG/509AF fits perfectly in applications ranging from wireless remote operation, security systems, to low-power remote transmitters. The EPROM technology makes customizing application programs (transmitter codes, appliance settings, etc.) extremely fast and convenient. The small footprint packages make this rfPIC™ perfect for applications with space limitations. Low-cost, low-power, high performance, ease of use and I/O flexibility make the rfPIC12C509AG/509AF very versatile.

**TABLE 1-1: rfPIC12C509AG/509AF DEVICE**

| | | rfPIC12C509AG | rfPIC12C509AF |
|---|---|---|---|
| Clock | Maximum Frequency of Operation (MHz) | 4 | |
| Memory | EPROM Program Memory | 1024 x 12 | |
| | RAM Data Memory (bytes) | 41 | |
| Peripherals | EEPROM Data Memory (bytes) | — | |
| | Timer Module(s) | TMR0 | |
| | A/D Converter (8-bit) Channels | — | |
| Features | Transmitter | ASK | ASK, FSK |
| | Wake-up from SLEEP on pin change | Yes | |
| | Interrupt Sources | — | |
| | I/O Pins | 5 | |
| | Input Pins | 1 | |
| | Internal Pull-ups | Yes | |
| | In-Circuit Serial Programming | Yes | |
| | Number of Instructions | 33 | |
| | Packages | 18-pin JW, SOIC | 20-pin JW, SSOP |

The rfPIC12C509AG/509AF has Power-on Reset, selectable Watchdog Timer, selectable code protect and high I/O current capability.
The rfPIC12C509AG/509AF has serial programming with data pin GP0 and clock pin GP1.

**NOTES:**

## 2.0 rfPIC12C509AG/509AF DEVICE VARIETIES

A variety of packaging options are available. Depending on application and production requirements, the proper device option can be selected using the information in this section. When placing orders, please use the rfPIC12C509AG/509AF Product Identification System at the back of this data sheet to specify the correct part number.

### 2.1 UV Erasable Devices

The UV erasable version, offered in a windowed ceramic DIP package, is optimal for prototype development and pilot programs.

The UV erasable version can be erased and reprogrammed to any of the configuration modes.

| | |
|---|---|
| **Note:** | Please note that erasing the device will also erase the pre-programmed internal calibration value for the internal oscillator. The calibration value must be saved prior to erasing the part. |

Microchip's PICSTART® PLUS and PRO MATE® programmers all support programming of the rfPIC12C509AG/509AF. Third party programmers also are available; refer to the *Microchip Third Party Guide* (DS00104) for a list of sources.

### 2.2 One-Time-Programmable (OTP) Devices

The availability of OTP devices is especially useful for customers who need the flexibility for frequent code updates or small volume applications.

The OTP devices, packaged in plastic packages permit the user to program them once. In addition to the program memory, the configuration bits must also be programmed.

### 2.3 Quick-Turnaround-Production (QTP)Devices

Microchip offers a QTP Programming Service for factory production orders. This service is made available for users who choose not to program a medium to high quantity of units and whose code patterns have stabilized. The devices are identical to the OTP devices but with all EPROM locations and fuse options already programmed by the factory. Certain code and prototype verification procedures do apply before production shipments are available. Please contact your local Microchip Technology sales office for more details.

### 2.4 Serialized Quick-Turnaround Production (SQTP℠) Devices

Microchip offers a unique programming service where a few user-defined locations in each device are programmed with different serial numbers. The serial numbers may be random, pseudo-random or sequential.

Serial programming allows each device to have a unique number which can serve as an entry-code, password or ID number.

**NOTES:**

## 3.0 ARCHITECTURAL OVERVIEW

The rfPIC12C509AG/509AF is a low-cost, high performance, 8-bit, fully static, EPROM-based CMOS microcontroller combined with a UHF ASK/FSK transmitter.

There are no internal electrical connections between the PICmicro MCU and the transmitter.

Section 7 has a detailed description of UHF ASK/FSK transmitter.

### 3.1 PICmicro Microcontroller Unit

The high performance of the rfPIC12C509AG/509AF family can be attributed to a number of architectural features commonly found in RISC microprocessors. To begin with, the rfPIC12C509AG/509AF uses a Harvard architecture in which program and data are accessed on separate buses. This improves bandwidth over traditional von Neumann architecture where program and data are fetched on the same bus. Separating program and data memory further allows instructions to be sized differently than the 8-bit wide data word. Instruction opcodes are 12 bits wide making it possible to have all single word instructions. A 12-bit wide program memory access bus fetches a 12-bit instruction in a single cycle. A two-stage pipeline overlaps fetch and execution of instructions. Consequently, all instructions (33) execute in a single cycle (1µs @ 4MHz) except for program branches.

The table below lists program memory (EPROM) and data memory (RAM) for each device.

| Device | Memory | | |
|---|---|---|---|
| | EPROM Program | RAM Data | Transmitter |
| rfPIC12C509AG | 1024 x 12 | 41 | ASK |
| rfPIC12C509AF | 1024 x 12 | 41 | ASK/FSK |

The rfPIC12C509AG/509AF can directly or indirectly address its register files and data memory. All special function registers including the program counter are mapped in the data memory. The rfPIC12C509AG/509AF has a highly orthogonal (symmetrical) instruction set that makes it possible to carry out any operation on any register using any addressing mode. This symmetrical nature and lack of special optimal situations make programming with the rfPIC12C509AG/509AF simple yet efficient. In addition, the learning curve is reduced significantly.

The rfPIC12C509AG/509AF contains an 8-bit ALU and working register. The ALU is a general purpose arithmetic unit. It performs arithmetic and Boolean functions between data in the working register and any register file.

The ALU is 8 bits wide and capable of addition, subtraction, shift and logical operations. Unless otherwise mentioned, arithmetic operations are two's complement in nature. In two-operand instructions, typically one operand is the W (working) register. The other operand is either a file register or an immediate constant. In single operand instructions, the operand is either the W register or a file register.

The W register is an 8-bit working register used for ALU operations. It is not an addressable register.

Depending on the instruction executed, the ALU may affect the values of the Carry (C), Digit Carry (DC), and Zero (Z) bits in the STATUS register. The C and DC bits operate as a borrow and digit borrow out bit, respectively, in subtraction. See the SUBWF and ADDWF instructions for examples.

### 3.2 UHF ASK/FSK Transmitter

The Transmitter is a fully integrated UHF ASK/FSK transmitter consisting of crystal oscillator, phase-locked loop (PLL), open-collector differential-output Power Amplifier (PA), and mode control logic. External components consist of bypass capacitors, crystal, and PLL loop filter. There are no internal electrical connections between the PICmicro MCU and the transmitter.

The rfPIC12C509AG is capable of Amplitude Shift Keying (ASK) modulation by turning the PA on and off. The rfPIC12C509AF is capable of ASK or Frequency Shift Keying (FSK) modulation by employing an internal FSK switch to pull the transmitter crystal via a second load capacitor.

The PICmicro MCU oscillator is independent from the transmitter crystal oscillator. The transmit frequency is fixed and set by an external reference crystal. Transmit frequencies in the range of 310 to 480 MHz can be selected. Output drive is an open-collector differential amplifier. The differential output is well suited for loop antennas. Output power is adjustable from +2 dBm to -12 dBm in six discrete steps.

The rfPIC12C509AG/509AF are radio frequency (RF) emitting devices. Wireless RF devices are governed by a country's regulating agency. For example, in the United States it is the Federal Communications Committee (FCC) and in Europe it is the European Conference of Postal and Telecommunications Administrations (CEPT). It is the responsibility of the designer to ensure that their end product conforms to rules and regulations of the country of use and/or sale.

RF devices require correct board level implementation in order to meet regulatory requirements. Layout considerations are given in Section 7.0 UHF ASK/FSK Transmitter.

A simplified block diagram is shown in Figure 3-1, with the corresponding device pins described in Table 3-1.

**FIGURE 3-1: rfPIC12C509AG/509AF BLOCK DIAGRAM**

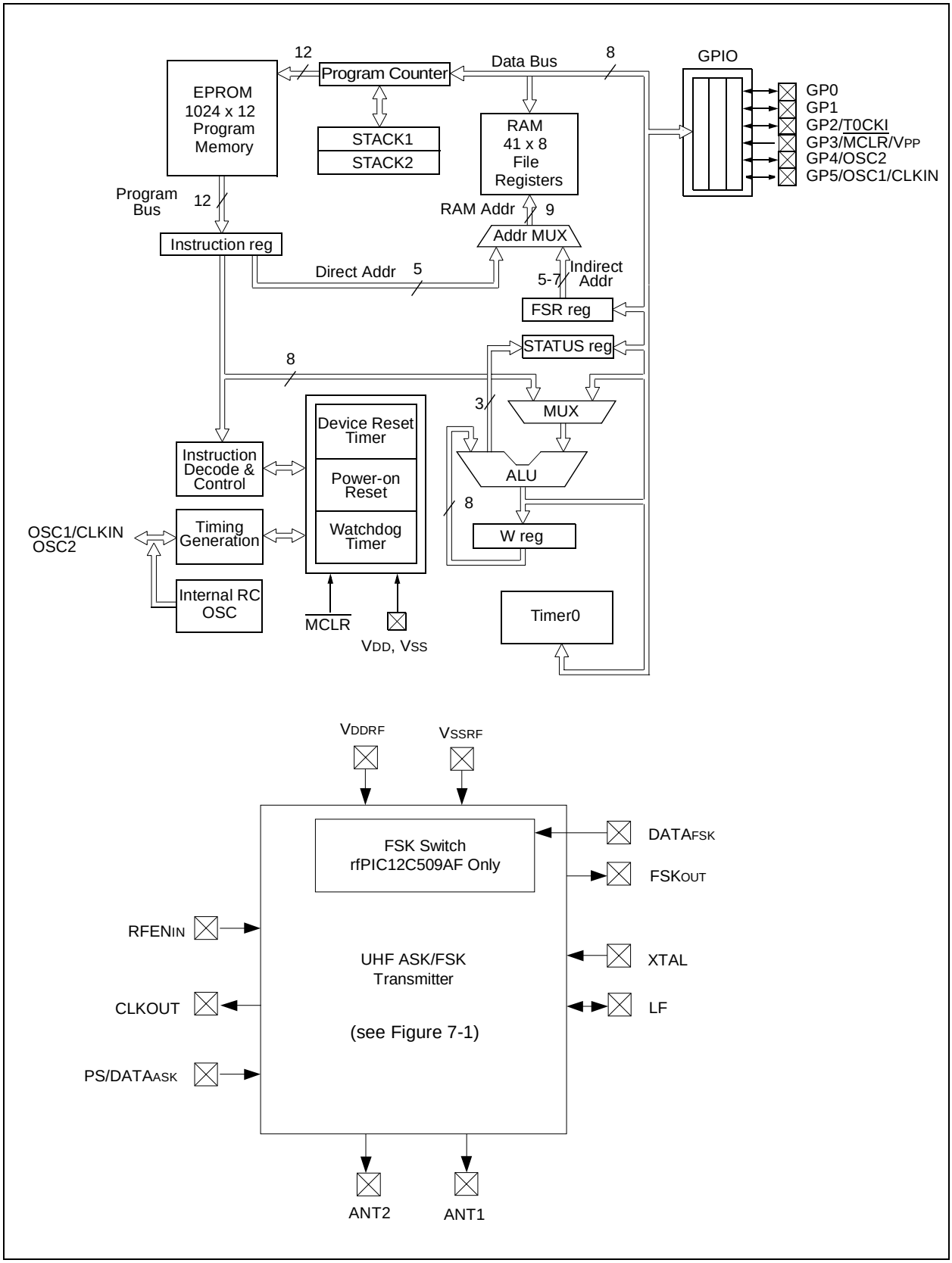


**TABLE 3-1: rfPIC12C509AG/509AF PINOUT DESCRIPTION**

| Name | SOIC CERDIP Pin # | SSOP Pin # | I/O/P Type | Buffer Type | Description |
|---|---|---|---|---|---|
| GP0 | 17 | 19 | I/O | TTL/ST | Bi-directional I/O port/ serial programming data. Can be software programmed for internal weak pull-up and wake-up from SLEEP on pin change. This buffer is a Schmitt Trigger input when used in serial programming mode. |
| GP1 | 16 | 18 | I/O | TTL/ST | Bi-directional I/O port/ serial programming clock. Can be software programmed for internal weak pull-up and wake-up from SLEEP on pin change. This buffer is a Schmitt Trigger input when used in serial programming mode. |
| GP2/T0CKI | 15 | 17 | I/O | ST | Bi-directional I/O port. Can be configured as T0CKI. |
| GP3/$\overline{\text{MCLR}}$/$V_{PP}$ | 4 | 4 | I | TTL/ST | Input port/master clear (Reset) input/programming voltage input. When configured as $\overline{\text{MCLR}}$, this pin is an active low RESET to the device. Voltage on $\overline{\text{MCLR}}$/$V_{PP}$ must not exceed $V_{DD}$ during normal device operation or the device will enter programming mode. Can be software programmed for internal weak pull-up and wake-up from SLEEP on pin change. Weak pull-up always on if configured as $\overline{\text{MCLR}}$. ST when in $\overline{\text{MCLR}}$ mode. |
| GP4/OSC2 | 3 | 3 | I/O | TTL | Bi-directional I/O port/oscillator crystal output. Connections to crystal or resonator in crystal oscillator mode (XT and LP modes only, GPIO in other modes). |
| GP5/OSC1/CLKIN | 2 | 2 | I/O | TTL/ST | Bi-directional IO port/oscillator crystal input/external clock source input (GPIO in Internal RC mode only, OSC1 in all other oscillator modes). TTL input when GPIO, ST input in external RC oscillator mode. |
| $V_{DD}$ | 1 | 1 | P | — | Positive supply for logic and I/O pins |
| $V_{SS}$ | 18 | 20 | P | — | Ground reference for logic and I/O pins |
| $RFEN_{IN}$ | 5 | 6 | I | TTL | Transmitter and CLKOUT enable. Internal pull-down. |
| CLKOUT | 6 | 7 | O | — | Clock output. |
| $PS/DATA_{ASK}$ | 7 | 8 | I | — | Power select and ASK data input. |
| $V_{DDRF}$ | 8 | 9 | P | — | Positive supply for transmitter. |
| ANT2 | 9 | 10 | O | — | Antenna connection to differential power amplifier output, open collector. |
| ANT1 | 10 | 11 | O | — | Antenna connection to differential power amplifier output, open collector. |
| $V_{SSRF}$ | 11 | 12 | P | — | Ground reference for transmitter. |
| LF | 13 | 14 | — | AN | External loop filter connection. Common node of charge pump output and VCO tuning input. |
| XTAL | 14 | 5 | I | — | Transmitter crystal connection to colpitts type crystal oscillator. |
| $DATA_{FSK}$ | — | 15 | I | TTL | FSK data input. |
| $FSK_{OUT}$ | — | 16 | O | — | FSK crystal pulling output. |

Legend: I = input, O = output, I/O = input/output, P = power, — = not used, TTL = TTL input, ST = Schmitt Trigger input, AN = analog, CMOS = CMOS

## 3.3 Clocking Scheme/Instruction Cycle

The clock input (OSC1/CLKIN pin) is internally divided by four to generate four non-overlapping quadrature clocks namely Q1, Q2, Q3 and Q4. Internally, the program counter is incremented every Q1, and the instruction is fetched from program memory and latched into instruction register in Q4. It is decoded and executed during the following Q1 through Q4. The clocks and instruction execution flow is shown in Figure 3-2 and Example 3-1.

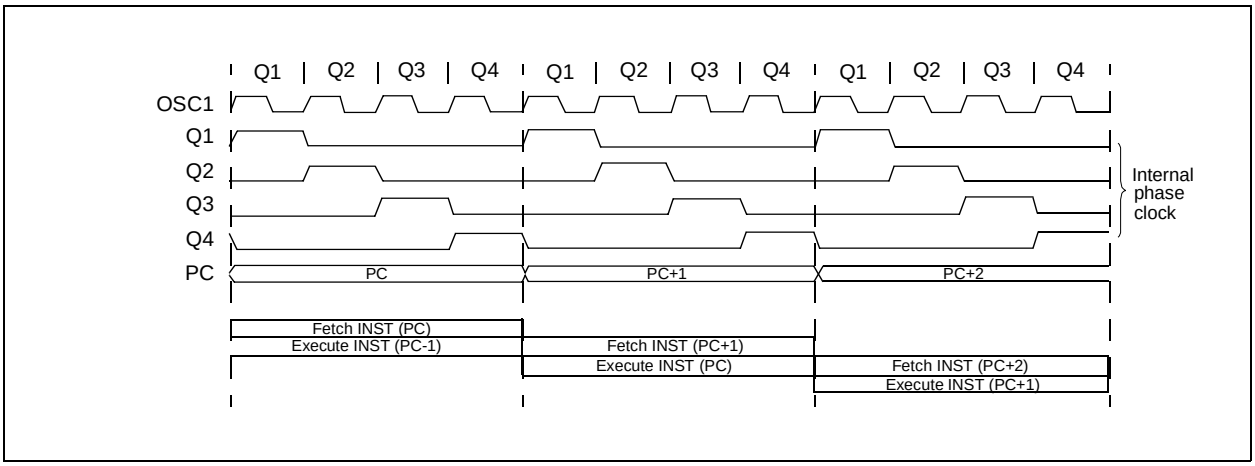
## 3.4 Instruction Flow/Pipelining

An Instruction Cycle consists of four Q cycles (Q1, Q2, Q3 and Q4). The instruction fetch and execute are pipelined such that fetch takes one instruction cycle while decode and execute takes another instruction cycle. However, due to the pipelining, each instruction effectively executes in one cycle. If an instruction causes the program counter to change (e.g., GOTO) then two cycles are required to complete the instruction (Example 3-1).

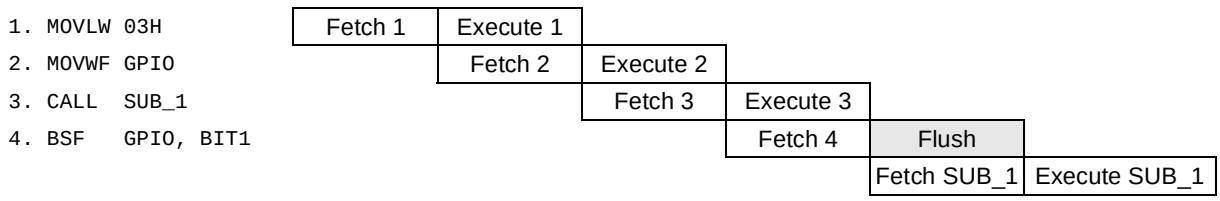
A fetch cycle begins with the program counter (PC) incrementing in Q1.

In the execution cycle, the fetched instruction is latched into the Instruction Register (IR) in cycle Q1. This instruction is then decoded and executed during the Q2, Q3, and Q4 cycles. Data memory is read during Q2 (operand read) and written during Q4 (destination write).

**FIGURE 3-2: CLOCK/INSTRUCTION CYCLE**




**EXAMPLE 3-1: INSTRUCTION PIPELINE FLOW**




All instructions are single cycle, except for any program branches. These take two cycles since the fetch instruction is “flushed” from the pipeline while the new instruction is being fetched and then executed.

**NOTES:**

## 4.0 MEMORY ORGANIZATION

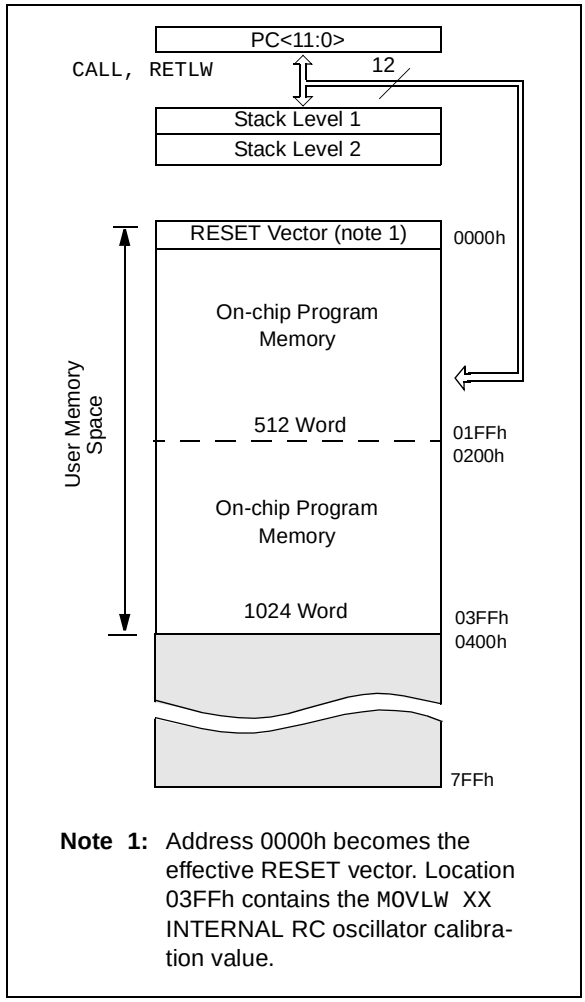
rfPIC12C509AG/509AF memory is organized into program memory and data memory. For devices with more than 512 bytes of program memory, a paging scheme is used. Program memory pages are accessed using one STATUS register bit. For the rfPIC12C509AG/509AF, with a data memory register file of more than 32 registers, a banking scheme is used. Data memory banks are accessed using the File Select Register (FSR).

### 4.1 Program Memory Organization

The rfPIC12C509AG/509AF devices have a 12-bit Program Counter (PC) capable of addressing a 2K x 12 program memory space.

Only the first 1K x 12 (0000h-03FFh) for the rfPIC12C509AG/509AF is physically implemented. Refer to Figure 4-1. Accessing a location above these boundaries will cause a wrap-around within the first 1K x 12 space. The effective RESET vector is at 000h, (see Figure 4-1). Location 03FFh contains the internal clock oscillator calibration value. This value should never be overwritten.

**FIGURE 4-1: PROGRAM MEMORY MAP AND STACK**




**Note 1:** Address 0000h becomes the effective RESET vector. Location 03FFh contains the MOVLW XX INTERNAL RC oscillator calibration value.

## 4.2 Data Memory Organization

Data memory is composed of registers, or bytes of RAM. Therefore, data memory for a device is specified by its register file. The register file is divided into two functional groups: special function registers and general purpose registers.

The special function registers include the TMR0 register, the Program Counter (PC), the Status Register, the I/O registers (ports), and the File Select Register (FSR). In addition, special purpose registers are used to control the I/O port configuration and prescaler options.

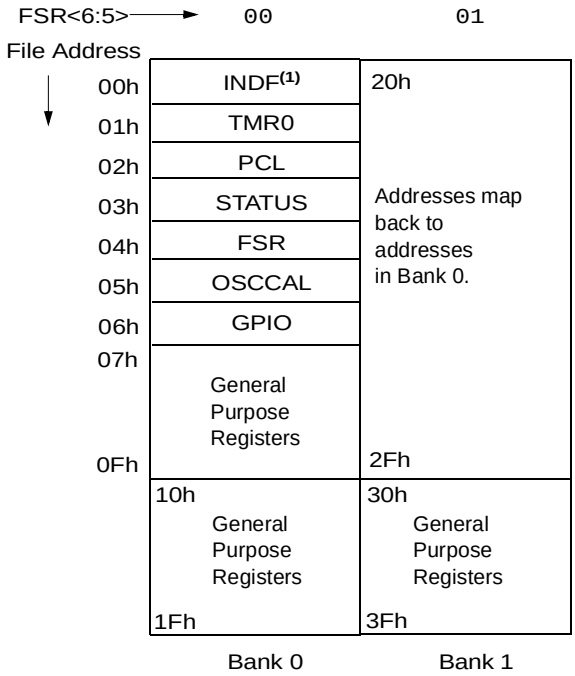
The general purpose registers are used for data and control information under command of the instructions.

For the rfPIC12C509AG/509AF, the register file is composed of 7 special function registers, 25 general purpose registers, and 16 general purpose registers that may be addressed using a banking scheme (Figure 4-2).

### 4.2.1 GENERAL PURPOSE REGISTER FILE

The general purpose register file is accessed either directly or indirectly through the file select register FSR

**FIGURE 4-2: rfPIC12C509AG/509AF REGISTER FILE MAP**




**Note 1:** Not a physical register. See Section 4.8

### 4.2.2 SPECIAL FUNCTION REGISTERS

The Special Function Registers (SFRs) are registers used by the CPU and peripheral functions to control the operation of the device (Table 4-1).

The special registers can be classified into two sets. The special function registers associated with the "core" functions are described in this section. Those related to the operation of the peripheral features are described in the section for each peripheral feature.

**TABLE 4-1: SPECIAL FUNCTION REGISTER (SFR) SUMMARY**

| Address | Name | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | Value on Power-on Reset | Value on All Other RESETS(2) |
|---|---|---|---|---|---|---|---|---|---|---|---|
| N/A | TRIS | — | — | | | | | | | --11 1111 | --11 1111 |
| N/A | OPTION | Contains control bits to configure Timer0, Timer0/WDT prescaler, wake-up on change, and weak pull-ups | | | | | | | | 1111 1111 | 1111 1111 |
| 00h | INDF | Uses contents of FSR to address data memory (not a physical register) | | | | | | | | xxxx xxxx | uuuu uuuu |
| 01h | TMR0 | 8-bit real-time clock/counter | | | | | | | | xxxx xxxx | uuuu uuuu |
| 02h(1) | PCL | Low order 8 bits of PC | | | | | | | | 1111 1111 | 1111 1111 |
| 03h | STATUS | GPWUF | — | PA0 | $\overline{TO}$ | $\overline{PD}$ | Z | DC | C | 0001 1xxx | q00q quuu(3) |
| 04h | FSR | Indirect data memory address pointer | | | | | | | | 110x xxxx | 11uu uuuu |
| 05h | OSCCAL | CAL5 | CAL4 | CAL3 | CAL2 | CAL1 | CAL0 | — | — | 1000 00-- | uuuu uu-- |
| 06h | GPIO | — | — | GP5 | GP4 | GP3 | GP2 | GP1 | GP0 | --xx xxxx | --uu uuuu |

Legend: Shaded boxes = unimplemented or unused, – = unimplemented, read as '0' (if applicable)
x = unknown, u = unchanged, q = see the tables in Section 8.7 for possible values.

**Note 1:** The upper byte of the Program Counter is not directly accessible. See Section 4.6 for an explanation of how to access these bits.

**2:** Other (non power-up) RESETS include external RESET through $\overline{MCLR}$, Watchdog Timer and Wake-up-on-Pin Change Reset.

**3:** If RESET was due to Wake-up-on-Pin Change then bit 7 = 1. All other RESETS will cause bit 7 = 0.

## 4.3 STATUS Register

This register contains the arithmetic status of the ALU, the RESET status, and the page preselect bit for program memories larger than 512 words.

The STATUS register can be the destination for any instruction, as with any other register. If the STATUS register is the destination for an instruction that affects the Z, DC or C bits, then the write to these three bits is disabled. These bits are set or cleared according to the device logic. Furthermore, the $\overline{TO}$ and $\overline{PD}$ bits are not writable. Therefore, the result of an instruction with the STATUS register as destination may be different than intended.

For example, `CLRF STATUS` will clear the upper three bits and set the Z bit. This leaves the STATUS register as `000u u1uu` (where u = unchanged).

It is recommended, therefore, that only `BCF`, `BSF` and `MOVWF` instructions be used to alter the STATUS register because these instructions do not affect the Z, DC or C bits from the STATUS register. For other instructions, which do affect STATUS bits, see Instruction Set Summary.

**figure 4-3: STATUS REGISTER (ADDRESS:03h)**

| R/W-0 | R/W-0 | R/W-0 | R-1 | R-1 | R/W-x | R/W-x | R/W-x |
|---|---|---|---|---|---|---|---|
| GPWUF | — | PA0 | $\overline{TO}$ | $\overline{PD}$ | Z | DC | C |
| bit7 | 6 | 5 | 4 | 3 | 2 | 1 | bit0 |

R = Readable bit
W = Writable bit
- n = Value at POR Reset

bit 7: **GPWUF**: GPIO Reset bit
1 = Reset due to wake-up from SLEEP on pin change
0 = After power-up or other RESET

bit 6: **Unimplemented**

bit 5: **PA0**: Program page preselect bits
1 = Page 1 (200h - 3FFh)
0 = Page 0 (000h - 1FFh
Each page is 512 bytes.
Using the PA0 bit as a general purpose read/write bit in devices which do not use it for program page preselect is not recommended since this may affect upward compatibility with future products.

bit 4: **$\overline{TO}$**: Time-out bit
1 = After power-up, `CLRWDT` instruction, or `SLEEP` instruction
0 = A WDT time-out occurred

bit 3: **$\overline{PD}$**: Power-down bit
1 = After power-up or by the `CLRWDT` instruction
0 = By execution of the `SLEEP` instruction

bit 2: **Z**: Zero bit
1 = The result of an arithmetic or logic operation is zero
0 = The result of an arithmetic or logic operation is not zero

bit 1: **DC**: Digit carry/$\overline{borrow}$ bit (for `ADDWF` and `SUBWF` instructions)
**ADDWF**
1 = A carry from the 4th low order bit of the result occurred
0 = A carry from the 4th low order bit of the result did not occur
**SUBWF**
1 = A borrow from the 4th low order bit of the result did not occur
0 = A borrow from the 4th low order bit of the result occurred

bit 0: **C**: Carry/$\overline{borrow}$ bit (for `ADDWF`, `SUBWF` and `RRF`, `RLF` instructions)

| **ADDWF** | **SUBWF** | **RRF or RLF** |
|---|---|---|
| 1 = A carry occurred | 1 = A borrow did not occur | Load bit with LSB or MSB, respectively |
| 0 = A carry did not occur | 0 = A borrow occurred | |

## 4.4 OPTION Register

The OPTION register is a 8-bit wide, write-only register which contains various control bits to configure the Timer0/WDT prescaler and Timer0.

By executing the `OPTION` instruction, the contents of the W register will be transferred to the OPTION register. A RESET sets the OPTION<7:0> bits.

| | |
|---|---|
| **Note:** | If TRIS bit is set to '0', the wake-up on change and pull-up functions are disabled for that pin (i.e., note that TRIS overrides OPTION control of $\overline{GPPU}$ and $\overline{GPWU}$). |
| **Note:** | If the T0CS bit is set to '1', GP2 is forced to be an input even if TRIS GP2 = '0'. |

**FIGURE 4-4: OPTION REGISTER**

| W-1 | W-1 | W-1 | W-1 | W-1 | W-1 | W-1 | W-1 |
|---|---|---|---|---|---|---|---|
| $\overline{GPWU}$ | $\overline{GPPU}$ | T0CS | T0SE | PSA | PS2 | PS1 | PS0 |
| bit7 | 6 | 5 | 4 | 3 | 2 | 1 | bit0 |

W = Writable bit
U = Unimplemented bit
- n = Value at POR Reset
Reference Table 4-1 for other RESETS.

bit 7: **$\overline{GPWU}$:** Enable wake-up on pin change (GP0, GP1, GP3)
1 = Disabled
0 = Enabled

bit 6: **$\overline{GPPU}$**: Enable weak pull-ups (GP0, GP1, GP3)
1 = Disabled
0 = Enabled

bit 5: **T0CS**: Timer0 clock source select bit
1 = Transition on T0CKI pin
0 = Transition on internal instruction cycle clock, Fosc/4

bit 4: **T0SE**: Timer0 source edge select bit
1 = Increment on high to low transition on the T0CKI pin
0 = Increment on low to high transition on the T0CKI pin

bit 3: **PSA**: Prescaler assignment bit
1 = Prescaler assigned to the WDT
0 = Prescaler assigned to Timer0

bit 2-0: **PS2:PS0**: Prescaler rate select bits

| Bit Value | Timer0 Rate | WDT Rate |
|---|---|---|
| 000 | 1 : 2 | 1 : 1 |
| 001 | 1 : 4 | 1 : 2 |
| 010 | 1 : 8 | 1 : 4 |
| 011 | 1 : 16 | 1 : 8 |
| 100 | 1 : 32 | 1 : 16 |
| 101 | 1 : 64 | 1 : 32 |
| 110 | 1 : 128 | 1 : 64 |
| 111 | 1 : 256 | 1 : 128 |

## 4.5 OSCCAL Register

The Oscillator Calibration (OSCCAL) register is used to calibrate the internal 4 MHz oscillator. It contains six bits for calibration. Increasing the cal value increases the frequency. See Section 8.2.5 for more information on the internal oscillator.

**FIGURE 4-5: OSCCAL REGISTER (ADDRESS 05h)**

| R/W-1 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | U-0 | U-0 |
|---|---|---|---|---|---|---|---|
| CAL5 | CAL4 | CAL3 | CAL2 | CAL1 | CAL0 | — | — |
| bit7 | | | | | | | bit0 |

R = Readable bit
W = Writable bit
U = Unimplemented bit, read as '0'
- n = Value at POR Reset

bit 7-2: **CAL<5:0>:** Calibration

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

## 4.6 Program Counter

As a program instruction is executed, the Program Counter (PC) will contain the address of the next program instruction to be executed. The PC value is increased by one every instruction cycle, unless an instruction changes the PC.

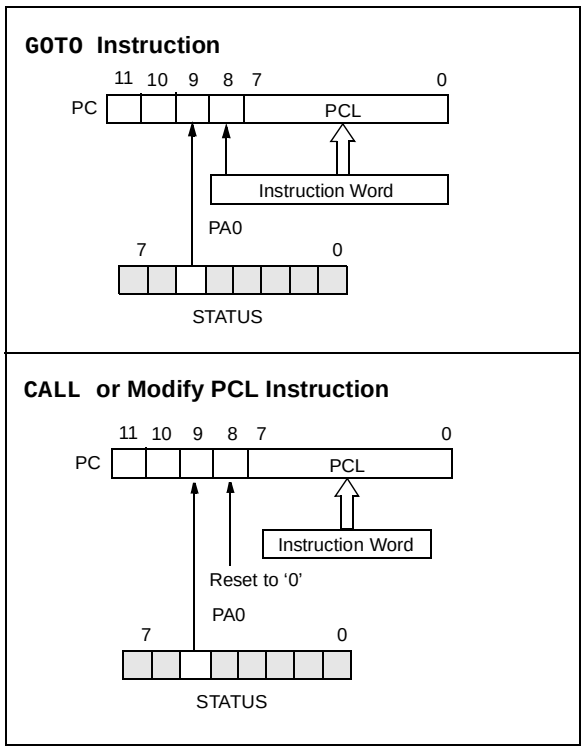
For a GOTO instruction, bits 8:0 of the PC are provided by the GOTO instruction word. The PC Latch (PCL) is mapped to PC<7:0>. Bit 5 of the STATUS register provides page information to bit 9 of the PC (Figure 4-6).

For a CALL instruction, or any instruction where the PCL is the destination, bits 7:0 of the PC again are provided by the instruction word. However, PC<8> does not come from the instruction word, but is always cleared (Figure 4-6).

Instructions where the PCL is the destination, or Modify PCL instructions, include MOVWF PC, ADDWF PC, and BSF PC,5.

| **Note:** | Because PC<8> is cleared in the CALL instruction, or any Modify PCL instruction, all subroutine calls or computed jumps are limited to the first 256 locations of any program memory page (512 words long). |
|---|---|

**FIGURE 4-6: LOADING OF PC BRANCH INSTRUCTIONS - rfPIC12C509AG/509AF**




### 4.6.1 EFFECTS OF RESET

The Program Counter is set upon a RESET, which means that the PC addresses the last location in the last page (i.e., the oscillator calibration instruction). After executing MOVLW XX, the PC will roll over to location 00h, and begin executing user code.

The STATUS register page preselect bits are cleared upon a RESET, which means that page 0 is preselected.

Therefore, upon a RESET, a GOTO instruction will automatically cause the program to jump to page 0 until the value of the page bits is altered.

## 4.7 Stack

The rfPIC12C509AG/509AF device has a 12-bit wide L.I.F.O. hardware push/pop stack.

A CALL instruction will *push* the current value of stack 1 into stack 2 and then push the current program counter value, incremented by one, into stack level 1. If more than two sequential CALL's are executed, only the most recent two return addresses are stored.

A RETLW instruction will *pop* the contents of stack level 1 into the program counter and then copy stack level 2 contents into level 1. If more than two sequential RETLW's are executed, the stack will be filled with the address previously stored in level 2. Note that the W register will be loaded with the literal value specified in the instruction. This is particularly useful for the implementation of data look-up tables within the program memory.

Upon any RESET, the contents of the stack remain unchanged, however the program counter (PCL) will also be Reset to 0.

| **Note 1:** | There are no STATUS bits to indicate stack overflows or stack underflow conditions. |
|---|---|
| **2:** | There are no instructions mnemonics called PUSH or POP. These are actions that occur from the execution of the CALL and RETLW instructions. |

## 4.8 Indirect Data Addressing; INDF and FSR Registers

The INDF register is not a physical register. Addressing INDF actually addresses the register whose address is contained in the FSR register (FSR is a *pointer*). This is indirect addressing.

**EXAMPLE 4-1: INDIRECT ADDRESSING**

- Register file 07 contains the value 10h
- Register file 08 contains the value 0Ah
- Load the value 07 into the FSR register
- A read of the INDF register will return the value of 10h
- Increment the value of the FSR register by one (FSR = 08)
- A read of the INDR register now will return the value of 0Ah.

Reading INDF itself indirectly (FSR = 0) will produce 00h. Writing to the INDF register indirectly results in a no-operation (although STATUS bits may be affected).

A simple program to clear RAM locations 10h-1Fh using indirect addressing is shown in Example 4-2.

**EXAMPLE 4-2: HOW TO CLEAR RAM USING INDIRECT ADDRESSING**

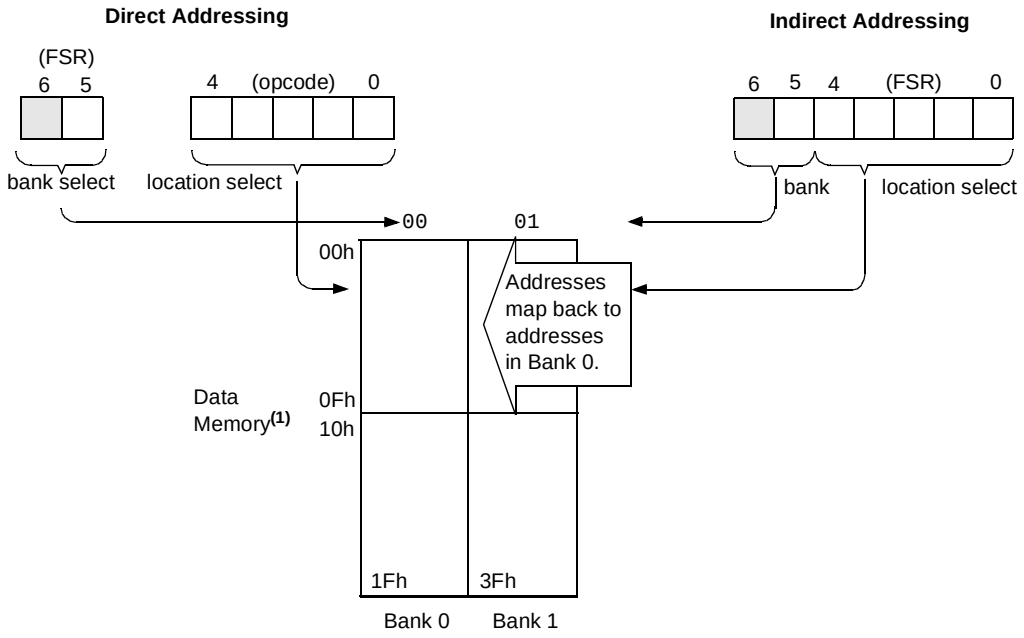
```
            movlw   0x10    ;initialize pointer
            movwf   FSR     ; to RAM
NEXT        clrf    INDF    ;clear INDF register
            incf    FSR,F   ;inc pointer
            btfsc   FSR,4   ;all done?
            goto    NEXT    ;NO, clear next
CONTINUE
            :               ;YES, continue
```

The FSR is a 5-bit wide register. It is used in conjunction with the INDF register to indirectly address the data memory area.

The FSR<4:0> bits are used to select data memory addresses 00h to 1Fh.

**rfPIC12C509AG/509AF:** Uses FSR<5>. Selects between bank 0 and bank 1. FSR<7:6> is unimplemented, read as '1'.

**FIGURE 4-7: DIRECT/INDIRECT ADDRESSING**




**Note 1:** For register map detail see Section 4.2.

## 5.0 I/O PORT

As with any other register, the I/O register can be written and read under program control. However, read instructions (e.g., `MOVF GPIO,W`) always read the I/O pins independent of the pin's input/output modes. On RESET, all I/O ports are defined as input (inputs are at hi-impedance) since the I/O control registers are all set.

### 5.1 GPIO

GPIO is an 8-bit I/O register. Only the low order 6 bits are used (GP5:GP0). Bits 7 and 6 are unimplemented and read as '0's. Please note that GP3 is an input only pin. The configuration word can set several I/O's to alternate functions. When acting as alternate functions the pins will read as '0' during port read. Pins GP0, GP1, and GP3 can be configured with weak pull-ups and also with wake-up on change. The wake-up on change and weak pull-up functions are not pin selectable. If pin 4 is configured as $\overline{MCLR}$, weak pull-up is always on and wake-up on change for this pin is not enabled.

### 5.2 TRIS Register

The output driver control register is loaded with the contents of the W register by executing the `TRIS f` instruction. A '1' from a TRIS register bit puts the corresponding output driver in a hi-impedance mode. A '0' puts the contents of the output data latch on the selected pins, enabling the output buffer. The exceptions are GP3 which is input only and GP2 which may be controlled by the option register, see Figure 4-4.

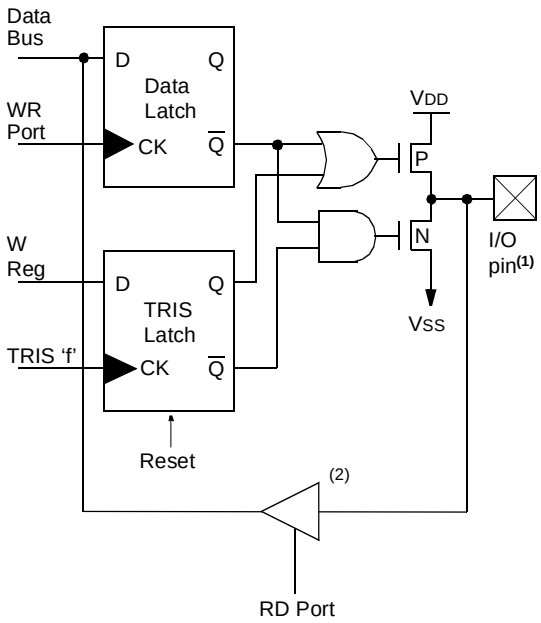
| | |
|---|---|
| **Note:** | A read of the ports reads the pins, not the output data latches. That is, if an output driver on a pin is enabled and driven high, but the external system is holding it low, a read of the port will indicate that the pin is low. |

The TRIS registers are "write-only" and are set (output drivers disabled) upon RESET.

### 5.3 I/O Interfacing

The equivalent circuit for an I/O port pin is shown in Figure 5-1. All port pins, except GP3 which is input only, may be used for both input and output operations. For input operations these ports are non-latching. Any input must be present until read by an input instruction (e.g., `MOVF GPIO,W`). The outputs are latched and remain unchanged until the output latch is rewritten. To use a port pin as output, the corresponding direction control bit in TRIS must be cleared (= 0). For use as an input, the corresponding TRIS bit must be set. Any I/O pin (except GP3) can be programmed individually as input or output.

**FIGURE 5-1: EQUIVALENT CIRCUIT FOR A SINGLE I/O PIN**




**Note 1:** I/O pins have protection diodes to $V_{DD}$ and $V_{SS}$.

**2:** See Table 3-1 for buffer type.

**TABLE 5-1: SUMMARY OF PORT REGISTERS**

| Address | Name | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | Value on Power-on Reset | Value on All Other RESETS |
|---|---|---|---|---|---|---|---|---|---|---|---|
| N/A | TRIS | — | — | | | | | | | --11 1111 | --11 1111 |
| N/A | OPTION | $\overline{GPWU}$ | $\overline{GPPU}$ | T0CS | T0SE | PSA | PS2 | PS1 | PS0 | 1111 1111 | 1111 1111 |
| 03H | STATUS | GPWUF | — | PAO | $\overline{TO}$ | $\overline{PD}$ | Z | DC | C | 0001 1xxx | q00q quuu(1) |
| 06h | GPIO | — | — | GP5 | GP4 | GP3 | GP2 | GP1 | GP0 | --xx xxxx | --uu uuuu |

Legend: Shaded cells not used by Port Registers, read as '0', — = unimplemented, read as '0', x = unknown, u = unchanged, q = see tables in Section 8.7 for possible values.

**Note 1:** If reset was due to wake-up on change, then bit 7 = 1. All other resets will cause bit 7 = 0.

## 5.4 I/O Programming Considerations

### 5.4.1 BI-DIRECTIONAL I/O PORTS

Some instructions operate internally as read followed by write operations. The BCF and BSF instructions, for example, read the entire port into the CPU, execute the bit operation and re-write the result. Caution must be used when these instructions are applied to a port where one or more pins are used as input/outputs. For example, a BSF operation on bit5 of GPIO will cause all eight bits of GPIO to be read into the CPU, bit5 to be set and the GPIO value to be written to the output latches. If another bit of GPIO is used as a bi-directional I/O pin (say bit0) and it is defined as an input at this time, the input signal present on the pin itself would be read into the CPU and rewritten to the data latch of this particular pin, overwriting the previous content. As long as the pin stays in the input mode, no problem occurs. However, if bit0 is switched into output mode later on, the content of the data latch may now be unknown.

Example 5-1 shows the effect of two sequential read-modify-write instructions (e.g., BCF, BSF, etc.) on an I/O port.

A pin actively outputting a high or a low should not be driven from external devices at the same time in order to change the level on this pin ("wired-or", "wired-and"). The resulting high output currents may damage the chip.

**EXAMPLE 5-1: Read-Modify-Write Instructions on an I/O Port**

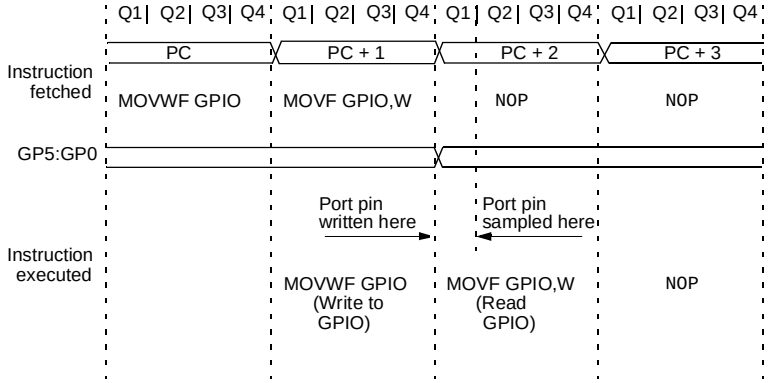
```
;Initial GPIO Settings
; GPIO<5:3> Inputs
; GPIO<2:0> Outputs
;
;                   GPIO latch  GPIO pins
;                   ----------  ----------
  BCF   GPIO, 5   ;--01 -ppp   --11 pppp
  BCF   GPIO, 4   ;--10 -ppp   --11 pppp
  MOVLW 007h       ;
  TRIS  GPIO      ;--10 -ppp   --11 pppp
;
;Note that the user may have expected the pin
;values to be --00 pppp. The 2nd BCF caused
;GP5 to be latched as the pin value (High).
```

### 5.4.2 SUCCESSIVE OPERATIONS ON I/O PORTS

The actual write to an I/O port happens at the end of an instruction cycle, whereas for reading, the data must be valid at the beginning of the instruction cycle (Figure 5-2). Therefore, care must be exercised if a write followed by a read operation is carried out on the same I/O port. The sequence of instructions should allow the pin voltage to stabilize (load dependent) before the next instruction, which causes that file to be read into the CPU, is executed. Otherwise, the previous state of that pin may be read into the CPU rather than the new state. When in doubt, it is better to separate these instructions with a NOP or another instruction not accessing this I/O port.

**FIGURE 5-2: SUCCESSIVE I/O OPERATION**




This example shows a write to GPIO followed by a read from GPIO.

Data setup time = (0.25 $T_{CY}$ – $T_{PD}$)

where: $T_{CY}$ = instruction cycle.

$T_{PD}$ = propagation delay

Therefore, at higher clock frequencies, a write followed by a read may be problematic.

**NOTES:**

## 6.0 TIMER0 MODULE AND TMR0 REGISTER

The Timer0 module has the following features:

- 8-bit timer/counter register, TMR0
  - Readable and writable
- 8-bit software programmable prescaler
- Internal or external clock select
  - Edge select for external clock

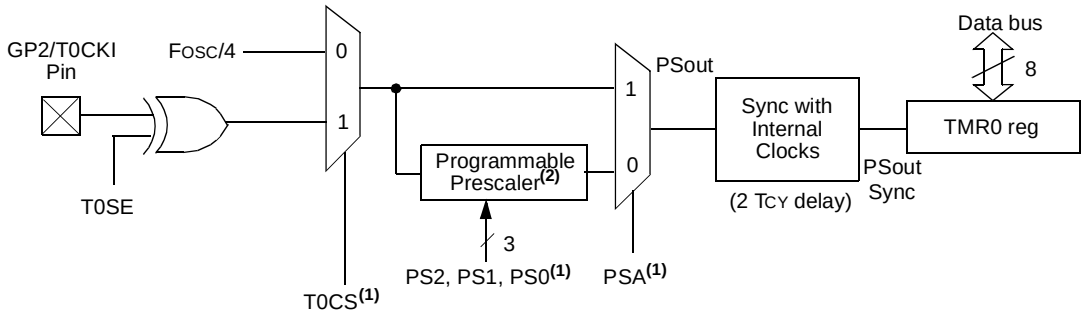
Figure 6-1 is a simplified block diagram of the Timer0 module.

Timer mode is selected by clearing the T0CS bit (OPTION<5>). In timer mode, the Timer0 module will increment every instruction cycle (without prescaler). If TMR0 register is written, the increment is inhibited for the following two instruction cycles (Figure 6-2 and Figure 6-3). The user can work around this by writing an adjusted value to the TMR0 register.

Counter mode is selected by setting the T0CS bit (OPTION<5>). In this mode, Timer0 will increment either on every rising or falling edge of pin T0CKI. The T0SE bit (OPTION<4>) determines the source edge. Clearing the T0SE bit selects the rising edge. Restrictions on the external clock input are discussed in detail in Section 6.1.

The prescaler may be used by either the Timer0 module or the Watchdog Timer, but not both. The prescaler assignment is controlled in software by the control bit PSA (OPTION<3>). Clearing the PSA bit will assign the prescaler to Timer0. The prescaler is not readable or writable. When the prescaler is assigned to the Timer0.

**FIGURE 6-1: TIMER0 BLOCK DIAGRAM**




**Note 1:** Bits T0CS, T0SE, PSA, PS2, PS1 and PS0 are located in the OPTION register.
**2:** The prescaler is shared with the Watchdog Timer (Figure 6-5).

**FIGURE 6-2: TIMER0 TIMING: INTERNAL CLOCK/NO PRESCALE**

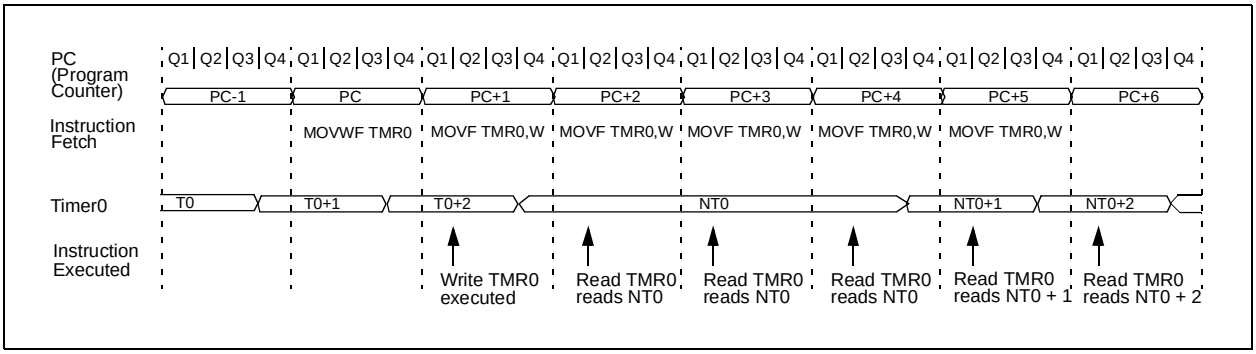



**FIGURE 6-3: TIMER0 TIMING: INTERNAL CLOCK/PRESCALE 1:2**

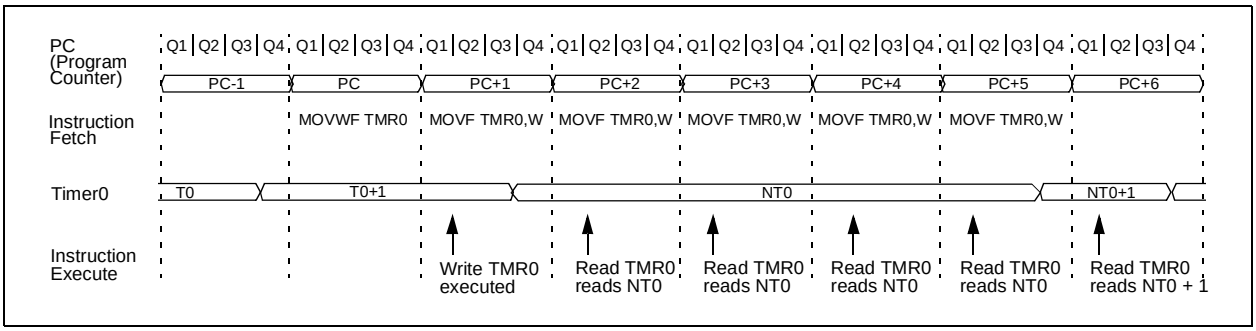



**TABLE 6-1: REGISTERS ASSOCIATED WITH TIMER0**

| Address | Name | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | Value on Power-on Reset | Value on All Other RESETS |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 01h | TMR0 | Timer0 - 8-bit real-time clock/counter | | | | | | | | xxxx xxxx | uuuu uuuu |
| N/A | OPTION | $\overline{\text{GPWU}}$ | $\overline{\text{GPPU}}$ | T0CS | T0SE | PSA | PS2 | PS1 | PS0 | 1111 1111 | 1111 1111 |
| N/A | TRIS | — | — | GP5 | GP4 | GP3 | GP2 | GP1 | GP0 | --11 1111 | --11 1111 |

Legend: Shaded cells not used by Timer0, - = unimplemented, x = unknown, u = unchanged,

## 6.1 Using Timer0 with an External Clock

When an external clock input is used for Timer0, it must meet certain requirements. The external clock requirement is due to internal phase clock (TOSC) synchronization. Also, there is a delay in the actual incrementing of Timer0 after synchronization.

### 6.1.1 EXTERNAL CLOCK SYNCHRONIZATION

When no prescaler is used, the external clock input is the same as the prescaler output. The synchronization of T0CKI with the internal phase clocks is accomplished by sampling the prescaler output on the Q2 and Q4 cycles of the internal phase clocks (Figure 6-4). Therefore, it is necessary for T0CKI to be high for at least 2TOSC (and a small RC delay of 20 ns) and low for at least 2TOSC (and a small RC delay of 20 ns). Refer to the electrical specification of the desired device.

When a prescaler is used, the external clock input is divided by the asynchronous ripple counter-type prescaler so that the prescaler output is symmetrical. For the external clock to meet the sampling requirement, the ripple counter must be taken into account. Therefore, it is necessary for T0CKI to have a period of at least 4TOSC (and a small RC delay of 40 ns) divided by the prescaler value. The only requirement on T0CKI high and low time is that they do not violate the minimum pulse width requirement of 10 ns. Refer to parameters 40, 41 and 42 in the electrical specification of the desired device.

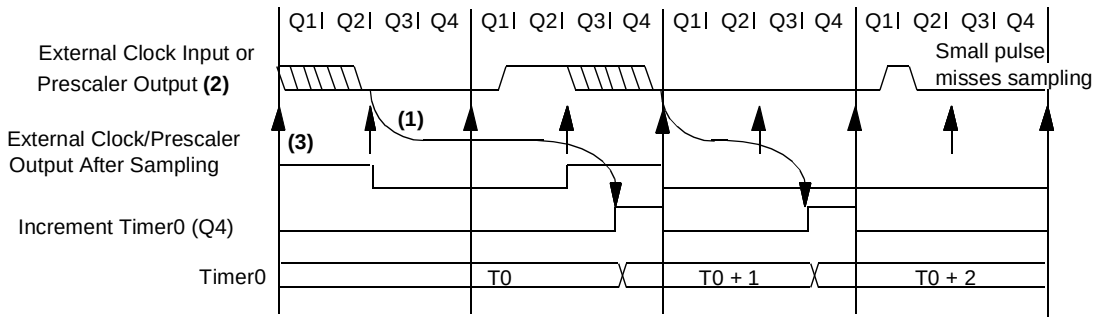
### 6.1.2 TIMER0 INCREMENT DELAY

Since the prescaler output is synchronized with the internal clocks, there is a small delay from the time the external clock edge occurs to the time the Timer0 module is actually incremented. Figure 6-4 shows the delay from the external clock edge to the timer incrementing.

### 6.1.3 OPTION REGISTER EFFECT ON GP2 TRIS

If the option register is set to read TIMER0 from the pin, the port is forced to an input regardless of the TRIS register setting.

**FIGURE 6-4: TIMER0 TIMING WITH EXTERNAL CLOCK**




**Note 1:** Delay from clock input change to Timer0 increment is 3Tosc to 7Tosc. (Duration of Q = Tosc). Therefore, the error in measuring the interval between two edges on Timer0 input = ± 4Tosc max.
**2:** External clock if no prescaler selected, Prescaler output otherwise.
**3:** The arrows indicate the points in time where sampling occurs.

## 6.2 Prescaler

An 8-bit counter is available as a prescaler for the Timer0 module, or as a postscaler for the Watchdog Timer (WDT), respectively (Section 8.6). For simplicity, this counter is being referred to as “prescaler” throughout this data sheet. Note that the prescaler may be used by either the Timer0 module or the WDT, but not both. Thus, a prescaler assignment for the Timer0 module means that there is no prescaler for the WDT, and vice-versa.

The PSA and PS2:PS0 bits (OPTION<3:0>) determine prescaler assignment and prescale ratio.

When assigned to the Timer0 module, all instructions writing to the TMR0 register (e.g., `CLRF 1`, `MOVWF 1`, `BSF 1,x`, etc.) will clear the prescaler. When assigned to WDT, a `CLRWDT` instruction will clear the prescaler along with the WDT. The prescaler is neither readable nor writable. On a RESET, the prescaler contains all '0's.

### 6.2.1 SWITCHING PRESCALER ASSIGNMENT

The prescaler assignment is fully under software control (i.e., it can be changed “on the fly” during program execution). To avoid an unintended device RESET, the following instruction sequence (Example 6-1) must be executed when changing the prescaler assignment from Timer0 to the WDT.

**EXAMPLE 6-1: Changing Prescaler (Timer0→WDT)**

```
1.CLRWDT              ;Clear WDT
2.CLRF   TMR0         ;Clear TMR0 & Prescaler
3.MOVLW  '00xx1111’b  ;These 3 lines (5, 6, 7)
4.OPTION              ; are required only if
                      ; desired
5.CLRWDT              ;PS<2:0> are 000 or 001
6.MOVLW  '00xx1xxx’b  ;Set Postscaler to
7.OPTION              ; desired WDT rate
```

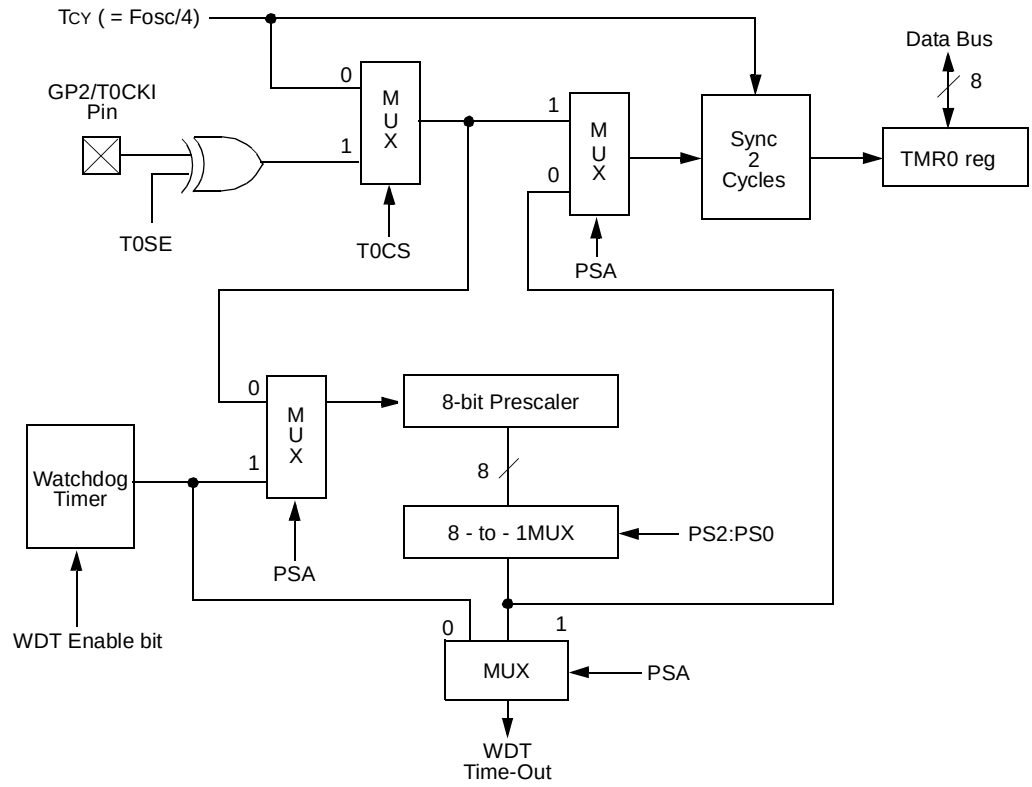
To change prescaler from the WDT to the Timer0 module, use the sequence shown in Example 6-2. This sequence must be used even if the WDT is disabled. A `CLRWDT` instruction should be executed before switching the prescaler.

**EXAMPLE 6-2: Changing Prescaler (WDT→Timer0)**

```
CLRWDT                ;Clear WDT and
                      ;prescaler
MOVLW   'xxxx0xxx'    ;Select TMR0, new
                      ;prescale value and
                      ;clock source
OPTION
```

**FIGURE 6-5: BLOCK DIAGRAM OF THE TIMER0/WDT PRESCALER**




**Note:** T0CS, T0SE, PSA, PS2:PS0 are bits in the OPTION register.

**NOTES:**

# 7.0 UHF ASK/FSK TRANSMITTER

## 7.1 Transmitter Operation

The transmitter is a fully integrated UHF ASK/FSK transmitter consisting of crystal oscillator, phase-locked loop (PLL), open-collector differential-output Power Amplifier (PA), and mode control logic. External components consist of bypass capacitors, crystal, and PLL loop filter. The rfPIC12C509AG is capable of Amplitude Shift Keying (ASK) modulation. The rfPIC12C509AF is capable of ASK or Frequency Shift Keying (FSK) modulation by employing an internal FSK switch to pull the transmitter crystal via a second load capacitor.

Figure 7-1 shows the internal structure of the transmitter. Transmitter connections are independent from the PICmicro microcontroller unit (MCU) to provide for maximum design flexibility. Example application circuits for ASK or FSK modulation are presented at the end of this section.

The rfPIC12C509AG/509AF are radio frequency (RF) emitting devices. Wireless RF devices are governed by a country's regulating agency. For example, in the United States it is the Federal Communications Committee (FCC) and in Europe it is the European Conference of Postal and Telecommunications Administrations (CEPT). It is the responsibility of the designer to ensure that their end product conforms to rules and regulations of the country of use and/or sale.

RF devices require correct board level implementation in order to meet regulatory requirements. Layout considerations are listed at the end of each subsection. It is best to place a ground plane on the PCB to reduce radio frequency emmissions and cross talk.

## 7.2 Supply Voltage ($V_{DDRF}$, $V_{SSRF}$)

Pins $V_{DDRF}$ and $V_{SSRF}$ supply power and ground respectively to the transmitter. These power pins are separate from power supply pins $V_{DD}$ and $V_{SS}$ to the PICmicro MCU.

Layout Considerations - Provide low impedance power and ground traces to minimize spurious emissions. A two-sided PCB with a ground plane on the bottom layer is highly recommended. Separate bypass capacitors should be connected as close as possible to each of the supply pins $V_{DD}$ and $V_{DDRF}$. Connect $V_{SS}$ and $V_{SSRF}$ to the ground plane using separate PCB vias. Do not share a PCB via with multiple ground traces.

## 7.3 Crystal Oscillator

The transmitter crystal oscillator is a Colpitts oscillator that provides the reference frequency to the PLL. It is independent from the PICmicro oscillator. An external crystal or AC coupled reference signal is connected to the XTAL pin. The transmit frequency is fixed and determined by the crystal frequency according to the formula:

$$f_{transmit} = f_{XTAL} \times 32$$

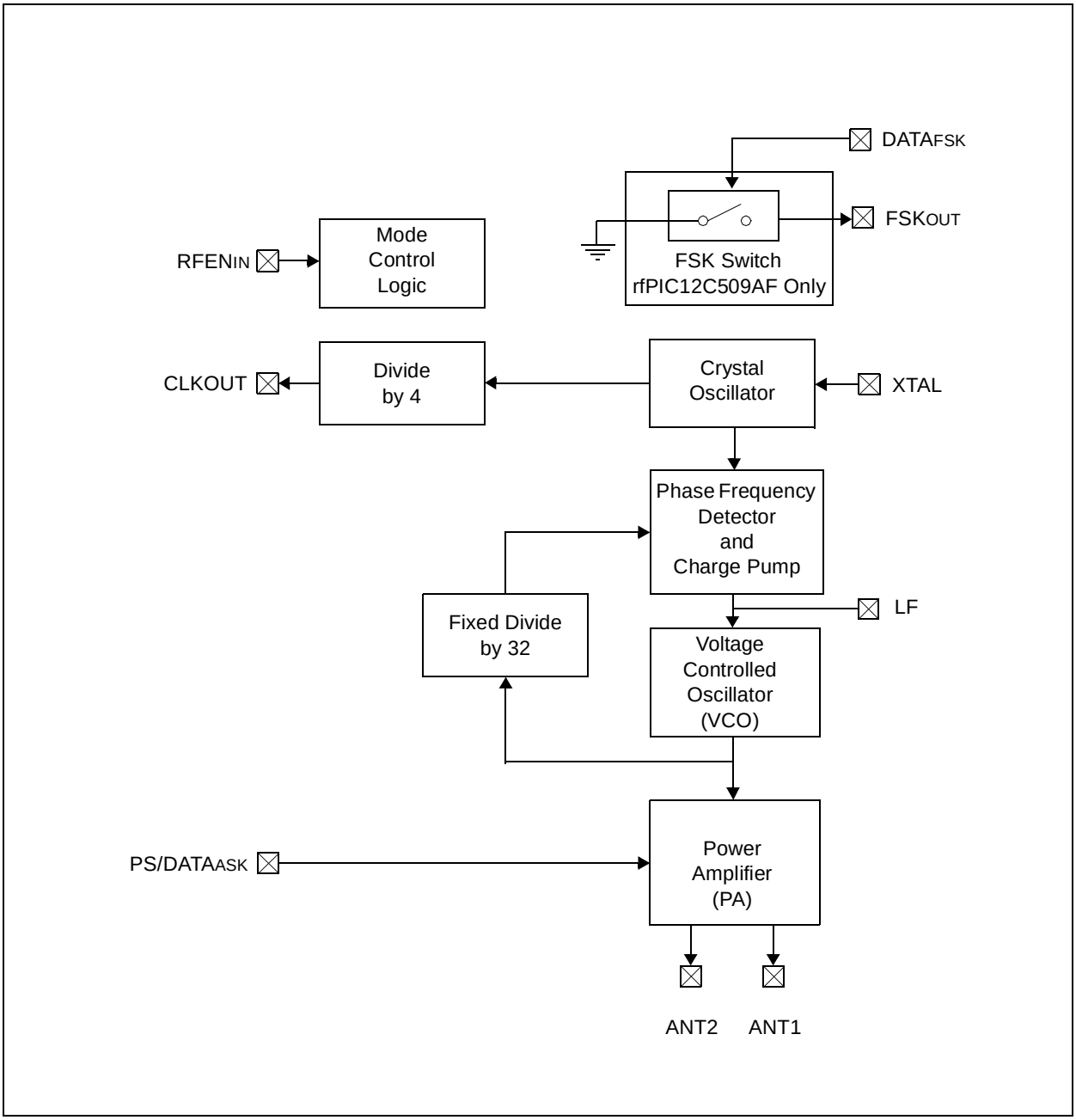
Due to the flexible selection of transmit frequency, the resulting crystal frequency may not be a standard off-the-shelf value. Therefore, for some carrier frequencies the designer will have to consult a crystal manufacturer and have a custom crystal manufactured. Crystal parameters are listed in Table 7-1. For background information on crystal selection see Application Note AN588, PICmicro® Microcontroller Oscillator Design Guide.

The crystal oscillator start time ($t_{on}$) is listed in Table 11-10, Transmitter AC Characteristics.

**TABLE 7-1: CRYSTAL PARAMETERS**

| Sym | Characteristic | Min | Max | Units | Conditions |
|---|---|---|---|---|---|
| $f_{XTAL}$ | Crystal Frequency | 9.69 | 15 | MHz | Parallel Resonant Mode |
| $C_L$ | Load Capacitance | 10 | 15 | pF | |
| $C_O$ | Shunt Capacitance | — | 7 | pF | |
| ESR | Equivalent Series Resistance | — | 60 | Ω | |
| These values are for design guidance only. | | | | | |

**FIGURE 7-1: rfPIC12C509AG/509AF TRANSMITTER BLOCK DIAGRAM**



### 7.3.1 CRYSTAL OSCILLATOR ASK OPERATION

The rfPIC12C509AG or 509AF crystal oscillator can be configured for ASK operation. Figure 7-2 shows an example ASK circuit.

Capacitor C1 trims the crystal load capacitance to the circuit load capacitance and places the crystal on the desired frequency.

**FIGURE 7-2: EXAMPLE ASK EXTERNAL CRYSTAL CIRCUIT**

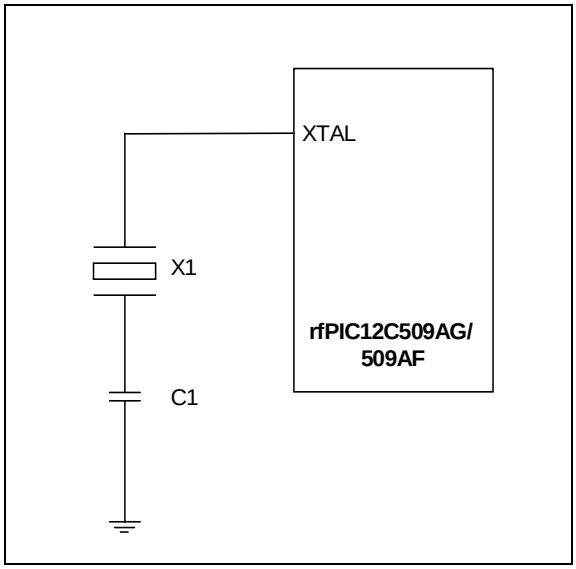



**TABLE 7-2: XTAL OSC APPROXIMATE FREQ. VS. CAPACITANCE (ASK MODE) (1)**

| C1 | Predicted Frequency (MHz) | PPM from 13.55 MHz | Transmit Frequency (MHz) (32 * $f_{XTAL}$) |
|---|---|---|---|
| 22 pF | 13.551438 | +106 | 433.646 |
| 39 pF | 13.550563 | +42 | 433.618 |
| 100 pF | 13.549844 | -12 | 433.595 |
| 150 pF | 13.549672 | -24 | 433.5895 |
| 470 pF | 13.549548 | -33 | 433.5856 |
| 1000 pF | 13.549344 | -48 | 433.579 |

**Note 1:** Standard Operating Conditions (unless otherwise stated) $T_A$ = 25°C, $RF_{EN}$ = 1, $V_{DDRF}$ = 3V, $f_{XTAL}$ = 13.55 MHz

### 7.3.2 CRYSTAL OSCILLATOR FSK OPERATION

The rfPIC12C509AF crystal oscillator can be configured for FSK operation. Figure 7-3 shows an example FSK circuit. Capacitors C1 and C2 achieve FSK modulation by pulling the crystal. When DATA$_{FSK}$ = 1, FSKOUT is high-impedance effectively coupling only capacitor C1 to the crystal and the resulting transmit frequency equals f$_{MAX}$. When DATA$_{FSK}$ = 0, FSKOUT is grounded to V$_{SSRF}$ and will parallel capacitor C2 with C1. The resulting transmit frequency will equal f$_{MIN}$.

Selecting the appropriate values for C1 and C2 sets the center frequency and frequency deviation. Capacitor C1 sets f$_{MAX}$ and capacitors C1 and C2 in parallel set f$_{MIN}$. The graph in Figure 7-4 illustrates this relationship. The transmit center frequency $f_C$ is defined as:

$$f_c = \frac{f_{max} + f_{min}}{2}$$

The frequency deviation of the transmit frequency is defined as:

$$\Delta f = \frac{f_{max} - f_{min}}{2}$$

Layout considerations - Avoid parallel traces in order to reduce circuit stray capacitance. Keep traces as short as possible. Isolate components to prevent coupling. Use ground traces to isolate signals.

**TABLE 7-3: TYPICAL TRANSMIT CENTER FREQUENCY AND FREQUENCY DEVIATION (FSK MODE) (1)**

| | C2 = 1000 pF | C2 = 100 pF | C2 = 47 pF |
|---|---|---|---|
| **C1 (pF)** | **Freq (MHz) / Dev (kHz)** | **Freq (MHz) / Dev (kHz)** | **Freq (MHz) / Dev (kHz)** |
| 22 | 433.612 / 34 | 433.619 / 27 | 433.625 / 21 |
| 33 | 433.604 / 25 | 433.610 / 19 | 433.614 / 14 |
| 39 | 433.598 / 20 | 433.604 / 14 | 433.608 / 10 |
| 47 | 433.596 / 17 | 433.601 / 11.5 | 433.604 / 8 |
| 68 | 433.593 / 13 | 433.598 / 9 | 433.600 / 5.5 |
| 100 | 433.587 / 8 | — | — |

**Note 1:** Standard Operating Conditions (unless otherwise stated) T$_A$ = 25°C, RF$_{EN}$ = 1, V$_{DDRF}$ = 3V, f$_{XTAL}$ = 13.55 MHz

**FIGURE 7-3: EXAMPLE FSK EXTERNAL CRYSTAL CIRCUIT**

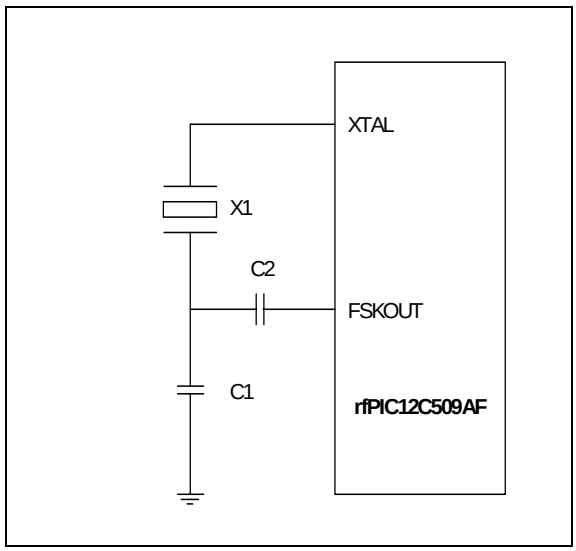
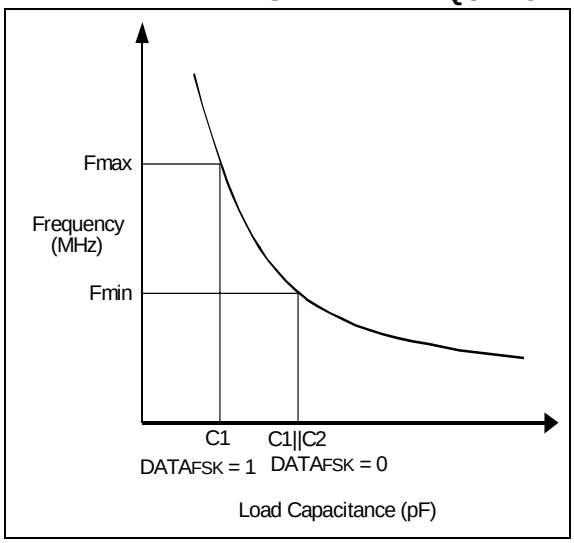



**FIGURE 7-4: LOAD CAPACITANCE VERSUS CHANGE IN TRANSMITTED FREQUENCY**



## 7.4 Clock Output (CLKOUT)

The crystal oscillator feeds a divide-by-four circuit that provides a clock output at the CLKOUT pin. The CLKOUT signal can be used as an input to the microcontroller or other external circuitry requiring a stable reference frequency. Do not connect the CLKOUT signal to the PICmicro OSC1 input because the PICmicro cannot run when there is no clock signal and therefore cannot enable the transmitter oscillator. It is required that the PICmicro be clocked externally or via the internal RC oscillator (see Section 8.2).

Connect CLKOUT to GP2/T0CKI input and use the Timer0 module if the application requires a stable reference frequency.

CLKOUT is slew-rate limited in order to keep spurious signal emissions as low as possible. The voltage swing ($V_{CLKOUT}$) depends on the capacitive loading ($C_{LOAD}$) on the CLKOUT pin (2 $V_{PP}$ at 5 pF).

Layout considerations - Shield each side of the clock output trace with ground traces to isolate the CLKOUT signal and reduce coupling.

## 7.5 Phase-Locked Loop (PLL)

The PLL consists of a phase-frequency detector (PFD), charge pump, voltage-controlled oscillator (VCO), and fixed divide-by-32 divider. An external loop filter is connected to pin LF. The loop filter controls the dynamic behavior of the PLL, primarily lock time and spur levels. The application determines the loop filter requirements.

The rfPIC™ employs a charge pump PLL that offers many advantages over the classical voltage phase detector PLL: infinite pull-in range and zero steady state phase error. The charge pump PLL allows the use of passive loop filters that are lower cost and minimize noise. Charge pump PLLs have reduced flicker noise thus limiting phase noise. Many of the classical texts on PLLs do not cover this type of PLL, however, today this is the most common type of PLL. This data sheet briefly covers the general terms and design requirements for the rfPIC. Detailed PLL design and operation is beyond the scope of this data sheet. For more information, the designer is referred to "*PLL Performance, Simulation, and Design*," Second Edition by Dean Banerjee ISBN 0970820704. Banerjee covers charge pump PLLs and loop filter selection.

The loop filter has a major impact on lock time and spur levels. Lock time is the time it takes the PLL to lock on frequency. When the PLL is first powered on or is changing frequencies, no data can be transmitted. Lock time must be considered before data transmission can begin. In addition to PLL lock time, the designer must take into account the crystal oscillator start time of approximately 1 ms. See Section 7.3 for more information about the crystal oscillator. Reference spurs occur at the carrier frequency plus and minus integer multiples of the reference frequency. Phase noise refers to noise generated by the PLL. Spur levels and phase noise can increase the signal to noise ratio (SNR) of the system and mask or degrade the transmitted signal.

The first order effect on PLL performance is loop bandwidth. Loop bandwidth ($\omega_c$) is defined as the point where the open loop phase transfer function equals 0 dB. Selecting a small loop bandwidth results in lower spur levels but slower lock time. Selecting a larger loop bandwidth results in a faster lock time but higher spur levels.

Second order effects on PLL performance is Phase margin ($\phi$) and Damping factor ($\zeta$). Phase margin is a measure of PLL stability. Choosing a phase margin that is too low will result in PLL instability. Choosing a higher phase margin results in less ringing and faster lock time at the expense of higher spur levels. Loop filters are typically designed for a total phase margin between 30 and 70 degrees. The aim of the designer is to choose a loop bandwidth and phase margin that gives the fastest possible lock time and meets the spur level requirements of the application.

Damping factor governs the second order transient response that determines the shape of the exponential envelope of the natural frequency. The natural frequency, also called ringing frequency, is the frequency of the VCO steering voltage as the PLL settles. Lock time is proportional to damping factor and inversely proportional to loop bandwidth.

The application determines the loop filter component requirements. For example, if the transmit frequency selected is near band edges or restricted bands, spur levels must be reduced to meet regulatory requirements. However, this will be at the expense of lock time. For an FSK application, a larger damping factor ($\cong$ 1.0) is desired so that there is less overshoot in the keying of FSK. For an ASK application, a damping factor = 0.707 results in less settling time and near optimum noise performance.

Figure 7-5 shows an example passive second order loop filter circuit. Table 7-4 gives example loop filter values for a crystal frequency of 13.56 MHz and transmit frequency of 433.92 MHz.

Layout considerations - Keep traces short and place loop filter components as close as possible to the LF pin.

**FIGURE 7-5: EXAMPLE LOOP FILTER CIRCUIT**

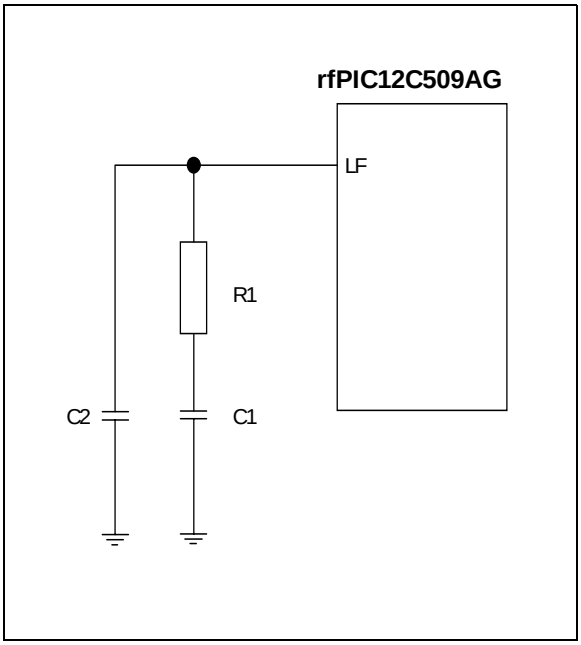



**TABLE 7-4: EXAMPLE LOOP FILTER VALUES (1)**

| C1 | C2 | R1 | Loop BW | Fn (natural freq in Hz) | Phase Margin (not counting sampling delay) | 2nd Order damping factor | Calculated Lock Time |
|---|---|---|---|---|---|---|---|
| 0.01 uF | 390 pF | 680 | 165 kHz | 64 kHz | 65 deg | 1.37 | 51 µS |
| 3900 pF | 100 pF | 1.5K | 360 kHz | 103 kHz | 63 deg | 1.89 | 16 µS |
| 1500 pF | 47 pF | 2.7K | 610 kHz | 166 kHz | 55 deg | 2.10 | 10 µS |
| 1000 pF | 18 pF | 4.7K | 1.05 MHz | 203 kHz | 50 deg | 3.0 | 8 µS |

**Note 1:** Standard Operating Conditions (unless otherwise stated) $T_A$ = 25°C, $RF_{EN}$ = 1, $V_{DDRF}$ = 3V, $f_{TRANSMIT}$ = 433.92 MHz

## 7.6 Power Amplifier

The PLL output feeds the power amplifier (PA). The open-collector differential output (ANT1, ANT2) can be used to drive a loop antenna directly or converted to single-ended output via an impenance matching network or balanced-to-unbalanced (balun) transformer. Pins ANT1 and ANT2 are open-collector outputs and must be pulled-up to $V_{DDRF}$ through the load.

The differential output of the PA should be matched to an impedance of 1 kΩ. Failure to match the impedance will cause excessive spurious and harmonic emissions.

The transmit output power can be adjusted in six discrete steps from +2 dBm to -12 dBm by varying the voltage ($V_{PS}$) at the PS/DATA$_{ASK}$ pin. Figure 7-6 shows an example voltage divider network for ASK operation and Figure 7-7 for FSK operation.

For FSK operation, the PS/DATA$_{ASK}$ pin only serves as a Power Select (PS) pin. An internal 20 µA current source pushes current through the PS/DATA$_{ASK}$ pin resulting in a voltage drop across resistor R2 at the $V_{PS}$ level selected for transmitter output power. $V_{PS}$ selects the PA bias current. Higher transmit power will draw higher current.

For ASK operation, the function of the PS/DATA$_{ASK}$ pin is to turn the Power Amplifier (PA) on and off. Resistors R1 and R2 form a voltage divider network to apply voltage $V_{PS}$ for the selected transmitter output power. If maximum transmitter output is desired, the output of a GP0 pin can be connected directly to PS/DATA$_{ASK}$.

Table 7-5 lists typical values for R1 and R2 for both the ASK and FSK modes.

**FIGURE 7-6: EXAMPLE ASK POWER SELECT CIRCUIT**

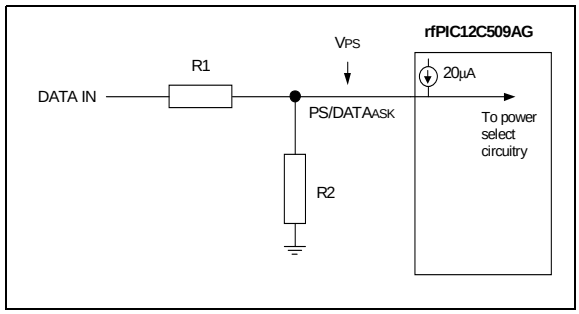



**FIGURE 7-7: EXAMPLE FSK POWER SELECT CIRCUIT**

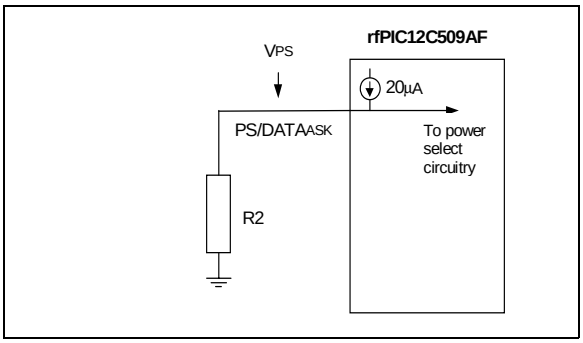



**TABLE 7-5: POWER SELECT (1)**

| Transmitter Output Power (dBm) | Transmitter Operating Current (mA) | Power Select (PS) Voltage $V_{PS}$ (Volts) (2) | ASK | | FSK |
|---|---|---|---|---|---|
| | | | R1 (Ω) | R2 (Ω) (3) | R2 (Ω) |
| +2 | 11.5 | ≥2.0 | 2400 | 4700 | ≥75K |
| -1 | 8.6 | 1.2 | 6800 | 4700 | 56K |
| -4 | 7.3 | 0.9 | 11K | 4700 | 47K |
| -7 | 6.2 | 0.7 | 15K | 4700 | 39K |
| -10 | 5.3 | 0.5 | 24K | 4700 | 27K |
| -12 | 4.8 | 0.3 | 43K | 4700 | 15K |
| -60 | <4.8 | <0.1 | OPEN | 4700 | 4700 |

**Note 1:** Standard Operating Conditions (unless otherwise stated) $T_A$ = 25°C, $RF_{EN}$ = 1, $V_{DDRF}$ = 3V, $f_{TRANSMIT}$ = 433.92 MHz

**2:** $V_{PS}$ is actual voltage on PS/DATA$_{ASK}$ pin.

**3:** The Power Select circuitry contains an internal 20 µA current source. To ensure that the transmitter output power is at the minimum when transmitting a DATA$_{ASK}$ = 0 ($V_{SSRF}$), select the value of resistor R2 such that the voltage drop across it is less than 0.1 volts.

## 7.7 Mode Control Logic

The mode control logic pin RFENIN controls the operation of the transmitter (Table 7-6). When RFENIN = 1 the transmitter and CLKOUT are enabled. When RFENIN = 0 the transmitter and CLKOUT are in standby mode. In standby mode the transmitter draws the least amount of current. The RFENIN pin has an internal pull-down resistor.

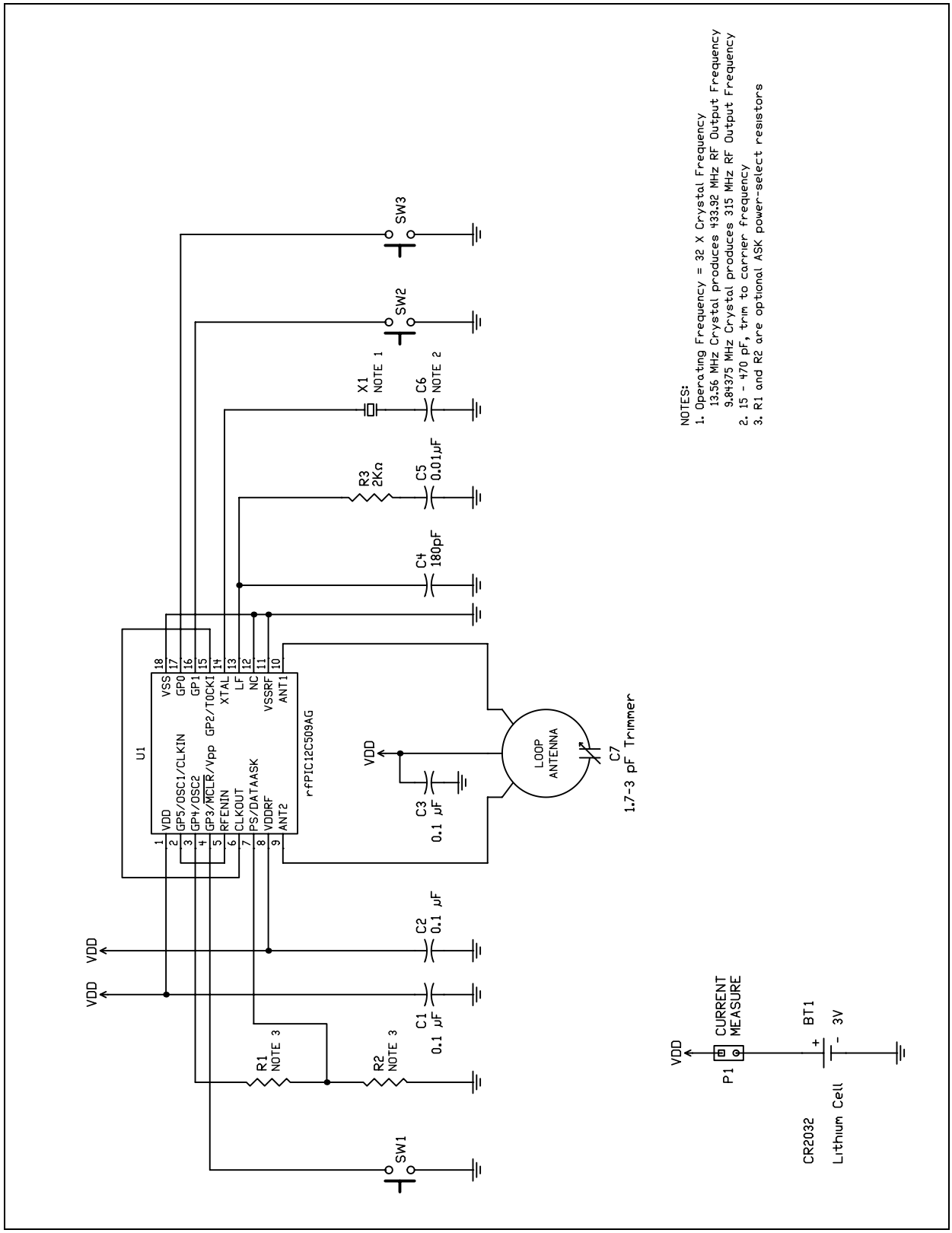
**TABLE 7-6: RFENIN PIN STATES**

| RFEN | Description |
|---|---|
| 0 | Transmitter and CLKOUT in Standby |
| 1 | Transmitter and CLKOUT enabled |

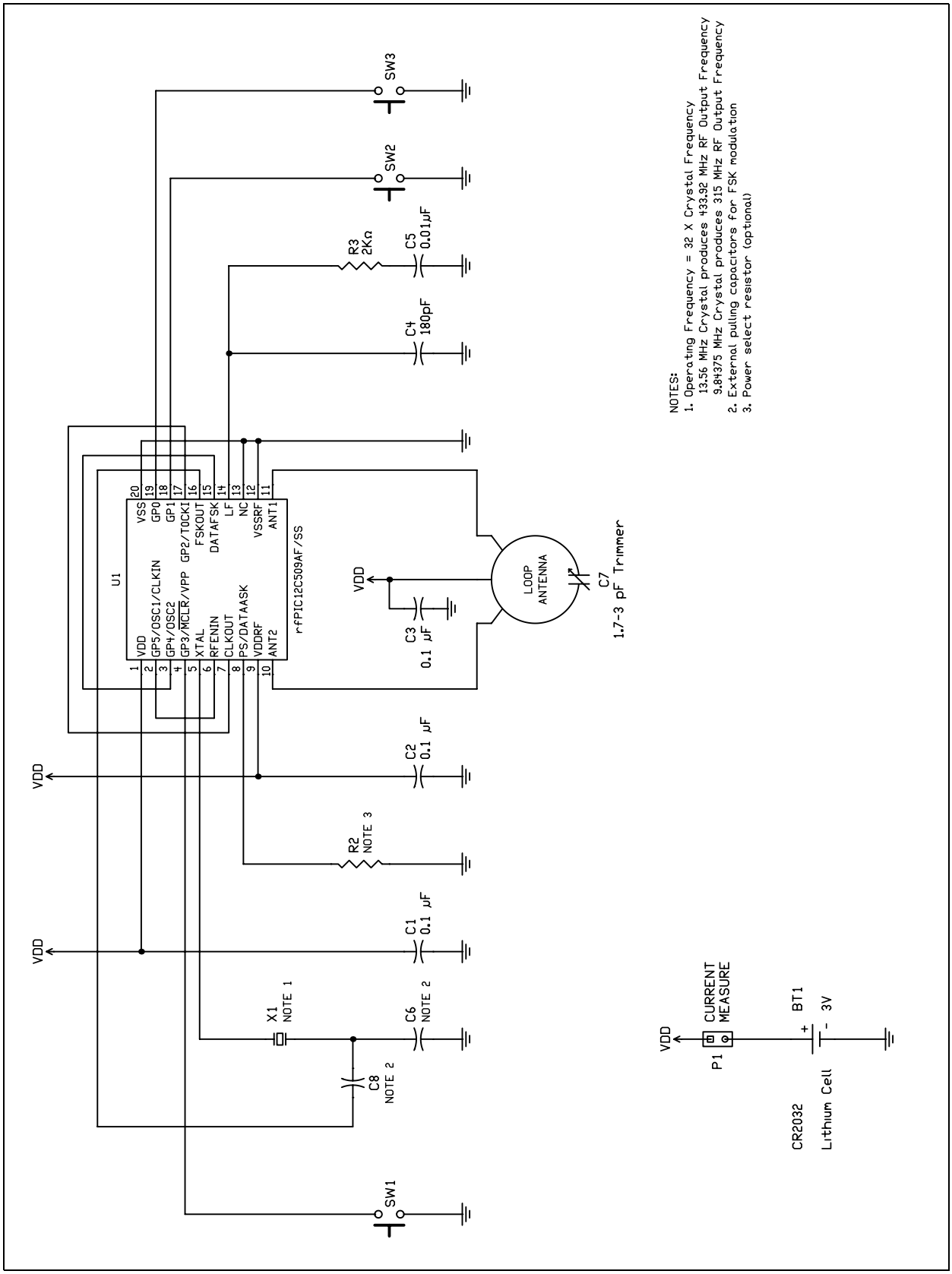
 

## 7.8 Application Circuits

### 7.8.1 EXAMPLE rfPIC12C509AG ASK CIRCUIT



7.8.2 EXAMPLE rfPIC12C509AF FSK CIRCUIT



## 8.0 SPECIAL FEATURES OF THE CPU

The rfPIC12C509AG/509AF microcontroller has a host of features intended to maximize system reliability, minimize cost through elimination of external components, provide power saving operating modes and offer code protection. These features are:

- Oscillator selection
- RESET
  - Power-on Reset (POR)
  - Device Reset Timer (DRT)
  - Wake-up from SLEEP on pin change
- Watchdog Timer (WDT)
- SLEEP
- Code protection
- ID locations
- In-Circuit Serial Programming

The above features are configured by the configuration bits during programming.

### 8.1 Configuration Bits

The rfPIC12C509AG/509AF configuration word consists of 12 bits. Configuration bits can be programmed to select various device configurations. Two bits are for the selection of the oscillator type, one bit is the Watchdog Timer enable bit, one bit for code protection, and one bit is the $\overline{MCLR}$ enable bit.

**FIGURE 8-1: CONFIGURATION WORD FOR rfPIC12C509AG/509AF**

| — | — | — | — | — | — | — | MCLRE | CP | WDTE | FOSC1 | FOSC0 |
|---|---|---|---|---|---|---|---|---|---|---|---|
| bit11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | bit0 |

| Register: | CONFIG |
|---|---|
| Address[1]: | FFFh |

bit 11-5: **Unimplemented**

bit 4: **MCLRE:** $\overline{MCLR}$ enable bit.
1 = $\overline{MCLR}$ pin enabled
0 = $\overline{MCLR}$ tied to $V_{DD}$, (Internally)

bit 3: **CP:** Code protection bit.
1 = Code protection off
0 = Code protection on

bit 2: **WDTE:** Watchdog timer enable bit
1 = WDT enabled
0 = WDT disabled

bit 1-0: **FOSC1:FOSC0:** Oscillator selection bits
11 = EXTRC - external RC oscillator
10 = INTRC - internal RC oscillator
01 = XT oscillator
00 = LP oscillator

**Note 1:** Refer to the PIC12C5XX Programming Specifications to determine how to access the configuration word. This register is not user addressable during device operation.

## 8.2 Oscillator Configurations

### 8.2.1 OSCILLATOR TYPES

The oscillator frequency is a primary factor in determining PICmicro microcontroller unit (MCU) current draw. As a rough guideline, the rfPIC12C509AG/509AF draws approximately 250 μA per MHz.

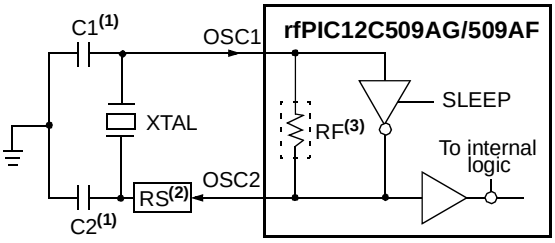
The rfPIC12C509AG/509AF can be operated in four different oscillator modes. The user can program two configuration bits (FOSC1:FOSC0) to select one of these four modes:

- LP: Low Power Crystal
- XT: Crystal/Resonator
- INTRC: Internal 4 MHz Oscillator
- EXTRC: External Resistor/Capacitor

### 8.2.2 CRYSTAL OSCILLATOR / CERAMIC RESONATORS

In XT or LP modes, a crystal or ceramic resonator is connected to the GP5/OSC1/CLKIN and GP4/OSC2 pins to establish oscillation (Figure 8-2). The rfPIC12C509AG/509AF oscillator design requires the use of a parallel cut crystal. Use of a series cut crystal may give a frequency out of the crystal manufacturers specifications. When in XT or LP modes, the device can have an external clock source drive the GP5/OSC1/CLKIN pin (Figure 8-3).

**FIGURE 8-2: CRYSTAL OPERATION (OR CERAMIC RESONATOR) (XT OR LP OSC CONFIGURATION)**




**Note 1:** See Capacitor Selection tables for recommended values of C1 and C2.
**2:** A series resistor (RS) may be required for AT strip cut crystals.
**3:** RF approximate value = 10 MΩ.

**FIGURE 8-3: EXTERNAL CLOCK INPUT OPERATION (XT OR LP OSC CONFIGURATION)**

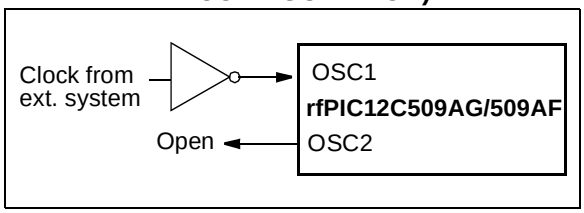



**TABLE 8-1: CAPACITOR SELECTION FOR CERAMIC RESONATORS - rfPIC12C509AG/509AF**

| Osc Type | Resonator Freq | Cap. Range C1 | Cap. Range C2 |
|---|---|---|---|
| XT | 4.0 MHz | 30 pF | 30 pF |

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 components.

**TABLE 8-2: CAPACITOR SELECTION FOR CRYSTAL OSCILLATOR - rfPIC12C509AG/509AF**

| Osc Type | Resonator Freq | Cap.Range C1 | Cap. Range C2 |
|---|---|---|---|
| LP | 32 kHz(1) | 15 pF | 15 pF |
| XT | 200 kHz<br>1 MHz<br>4 MHz | 47-68 pF<br>15 pF<br>15 pF | 47-68 pF<br>15 pF<br>15 pF |

**Note 1:** For $V_{DD}$ > 4.5V, C1 = C2 ≈ 30 pF is recommended.

These values are for design guidance only. Rs may be required 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.

### 8.2.3 EXTERNAL RC OSCILLATOR

For timing insensitive applications, the RC 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 and C components used.

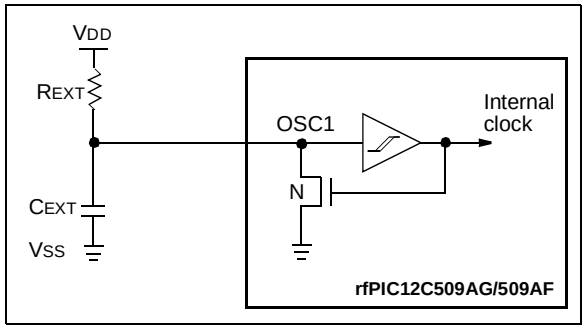
Figure 8-4 shows how the R/C combination is connected to the rfPIC12C509AG/509AF. For $R_{EXT}$ values below 2.2 kΩ, the oscillator operation may become unstable, or stop completely. For very high $R_{EXT}$ values (e.g., 1 MΩ) the oscillator becomes sensitive to noise, humidity and leakage. Thus, we recommend keeping $R_{EXT}$ between 3 kΩ and 100 kΩ.

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 or package lead frame capacitance.

The Electrical Specifications sections show RC frequency variation from part to part due to normal process variation. The variation is larger for larger R (since leakage current variation will affect RC frequency more for large R) and for smaller C (since variation of input capacitance will affect RC frequency more).

Also, see the Electrical Specifications sections 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, C, and $V_{DD}$ values.

**FIGURE 8-4: EXTERNAL RC OSCILLATOR MODE**



### 8.2.4 INTERNAL 4 MHz RC OSCILLATOR

The internal RC oscillator provides a fixed 4 MHz (nominal) system clock at $V_{DD}$ = 5V and 25°C, see "Electrical Specifications" section for information on variation over voltage and temperature.

In addition, a calibration instruction is programmed into the top of memory which contains the calibration value for the internal RC oscillator. This location is never code protected regardless of the code protect settings. This value is programmed as a `MOVLW XX` instruction where XX is the calibration value, and is placed at the RESET vector. This will load the W register with the calibration value upon RESET and the PC will then roll over to the users program at address 0x000. The user then has the option of writing the value to the OSCCAL Register (05h) or ignoring it.

OSCCAL, when written to with the calibration value, will "trim" the internal oscillator to remove process variation from the oscillator frequency.

| | |
|---|---|
| **Note:** | Please note that erasing the device will also erase the pre-programmed internal calibration value for the internal oscillator. The calibration value must be read prior to erasing the part. so it can be reprogrammed correctly later. |

For the rfPIC12C509AG/509AF bits <7:2>, CAL5-CAL0 are used for calibration. Adjusting CAL5-0 from 000000 to 111111 yields a higher clock speed. Note that bits 1 and 0 of OSCCAL are unimplemented and should be written as 0 when modifying OSCCAL for compatibility with future devices.

## 8.3 RESET

The device differentiates between various kinds of RESET:

a) Power-on Reset (POR)
b) $\overline{\text{MCLR}}$ Reset during normal operation
c) $\overline{\text{MCLR}}$ Reset during SLEEP
d) WDT Time-out Reset during normal operation
e) WDT Time-out Reset during SLEEP
f) Wake-up from SLEEP on pin change

Some registers are not RESET in any way; they are unknown on POR and unchanged in any other RESET. Most other registers are RESET to "RESET state" on Power-on Reset (POR), $\overline{\text{MCLR}}$, WDT or Wake-up-on-Pin Change Reset during normal operation. They are not affected by a WDT Reset during SLEEP or $\overline{\text{MCLR}}$ Reset during SLEEP, since these RESETS are viewed as resumption of normal operation. The exceptions to this are $\overline{\text{TO}}$, $\overline{\text{PD}}$, and GPWUF bits. They are set or cleared differently in different RESET situations. These bits are used in software to determine the nature of reset. See Table 8-3 for a full description of RESET states of all registers.

**TABLE 8-3: RESET CONDITIONS FOR REGISTERS**

| Register | Address | Power-on Reset | $\overline{\text{MCLR}}$ Reset<br>WDT time-out<br>Wake-up-on-Pin Change |
|---|---|---|---|
| W | — | qqqq qqxx (1) | qqqq qquu (1) |
| INDF | 00h | xxxx xxxx | uuuu uuuu |
| TMR0 | 01h | xxxx xxxx | uuuu uuuu |
| PC | 02h | 1111 1111 | 1111 1111 |
| STATUS | 03h | 0001 1xxx | q00q quuu (2,3) |
| FSR | 04h | 110x xxxx | 11uu uuuu |
| OSCCAL | 05h | 1000 00-- | uuuu uu-- |
| GPIO | 06h | --xx xxxx | --uu uuuu |
| OPTION | — | 1111 1111 | 1111 1111 |
| TRIS | — | --11 1111 | --11 1111 |

Legend: u = unchanged, x = unknown, - = unimplemented bit, read as '0', q = value depends on condition.
**Note 1:** Bits <7:2> of W register contain oscillator calibration values due to MOVLW XX instruction at top of memory.
**Note 2:** See Table 8-7 for RESET value for specific conditions
**Note 3:** If RESET was due to Wake-up-on-Pin Change, then bit 7 = 1. All other RESETS will cause bit 7 = 0.

**TABLE 8-4: RESET CONDITION FOR SPECIAL REGISTERS**

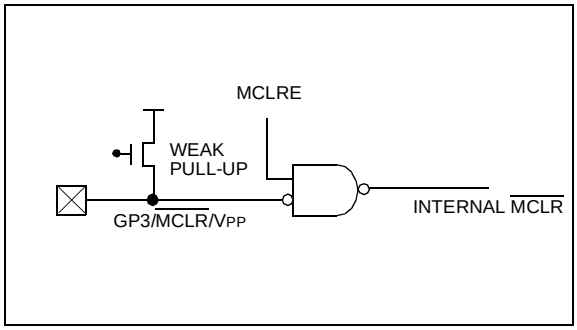
| | STATUS Addr: 03h | PCL Addr: 02h |
|---|---|---|
| Power-on Reset | 0001 1xxx | 1111 1111 |
| $\overline{\text{MCLR}}$ Reset during normal operation | 000u uuuu | 1111 1111 |
| $\overline{\text{MCLR}}$ Reset during SLEEP | 0001 0uuu | 1111 1111 |
| WDT Reset during SLEEP | 0000 0uuu | 1111 1111 |
| WDT Reset normal operation | 0000 uuuu | 1111 1111 |
| Wake-up from SLEEP on pin change | 1001 0uuu | 1111 1111 |

Legend: u = unchanged, x = unknown, - = unimplemented bit, read as '0'.

### 8.3.1 MCLR ENABLE

This configuration bit when unprogrammed (left in the '1' state) enables the external MCLR function. When programmed, the MCLR function is tied to the internal VDD, and the pin is assigned to be a GPIO. See Figure 8-5. When pin GP3/MCLR/VPP is configured as MCLR, the internal pull-up is always on.

**FIGURE 8-5: MCLR SELECT**




## 8.4 Power-On Reset (POR)

The rfPIC12C509AG/509AF incorporates on-chip Power-on Reset (POR) circuitry which provides an internal chip RESET for most power-up situations.

The on-chip POR circuit holds the chip in RESET until VDD has reached a high enough level for proper operation. To take advantage of the internal POR, program the GP3/MCLR/VPP pin as MCLR and tie through a resistor to VDD or program the pin as GP3. An internal weak pull-up resistor is implemented using a transistor. Refer to Table 11-1 for the pull-up resistor ranges. This will eliminate external RC components usually needed to create a Power-on Reset. A maximum rise time for VDD is specified. See Electrical Specifications for details.

When the device starts normal operation (exits the RESET condition), device operating parameters (voltage, frequency, temperature,...) must be met to ensure operation. If these conditions are not met, the device must be held in RESET until the operating parameters are met.

A simplified block diagram of the on-chip Power-on Reset circuit is shown in Figure 8-6.

The Power-on Reset circuit and the Device Reset Timer (Section 8.5) circuit are closely related. On power-up, the RESET latch is set and the DRT is RESET. The DRT timer begins counting once it detects MCLR to be high. After the time-out period, which is typically 18 ms, it will RESET the RESET latch and thus end the on-chip RESET signal.

A power-up example where MCLR is held low is shown in Figure 8-7. VDD is allowed to rise and stabilize before bringing MCLR high. The chip will actually come out of RESET TDRT msec after MCLR goes high.

In Figure 8-8, the on-chip Power-on Reset feature is being used (MCLR and VDD are tied together or the pin is programmed to be GP3.). The VDD is stable before the start-up timer times out and there is no problem in getting a proper RESET. However, Figure 8-9 depicts a problem situation where VDD rises too slowly. The time between when the DRT senses that MCLR is high and when MCLR (and VDD) actually reach their full value, is too long. In this situation, when the start-up timer times out, VDD has not reached the VDD (min) value and the chip is, therefore, not guaranteed to function correctly. For such situations, we recommend that external RC circuits be used to achieve longer POR delay times (Figure 8-8).

| | |
|---|---|
| **Note:** | When the device starts normal operation (exits the RESET condition), device operating parameters (voltage, frequency, temperature, etc.) must be met to ensure operation. If these conditions are not met, the device must be held in RESET until the operating conditions are met. |

For additional information refer to Application Notes "*Power-Up Considerations*" - AN522 and "*Power-up Trouble Shooting*" - AN607.

**FIGURE 8-6: SIMPLIFIED BLOCK DIAGRAM OF ON-CHIP RESET CIRCUIT**

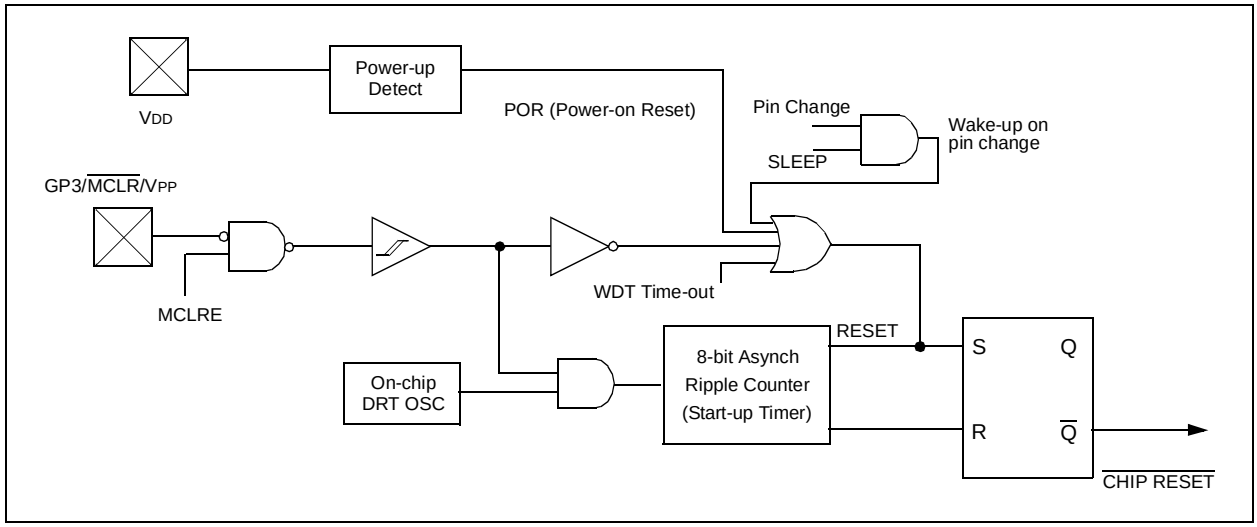



**FIGURE 8-7: TIME-OUT SEQUENCE ON POWER-UP ($\overline{MCLR}$ PULLED LOW)**

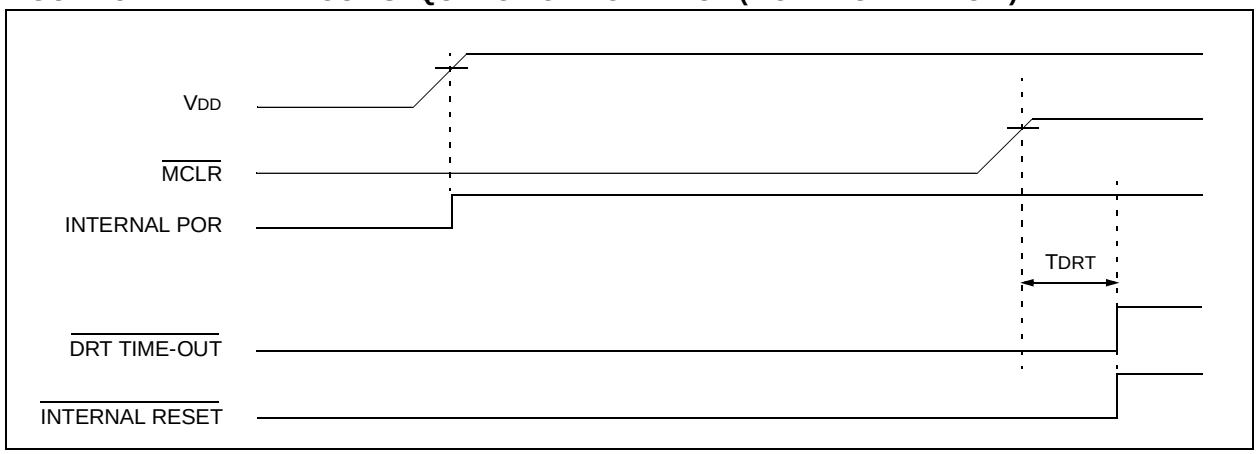



**FIGURE 8-8: TIME-OUT SEQUENCE ON POWER-UP ($\overline{MCLR}$ TIED TO VDD): FAST VDD RISE TIME**

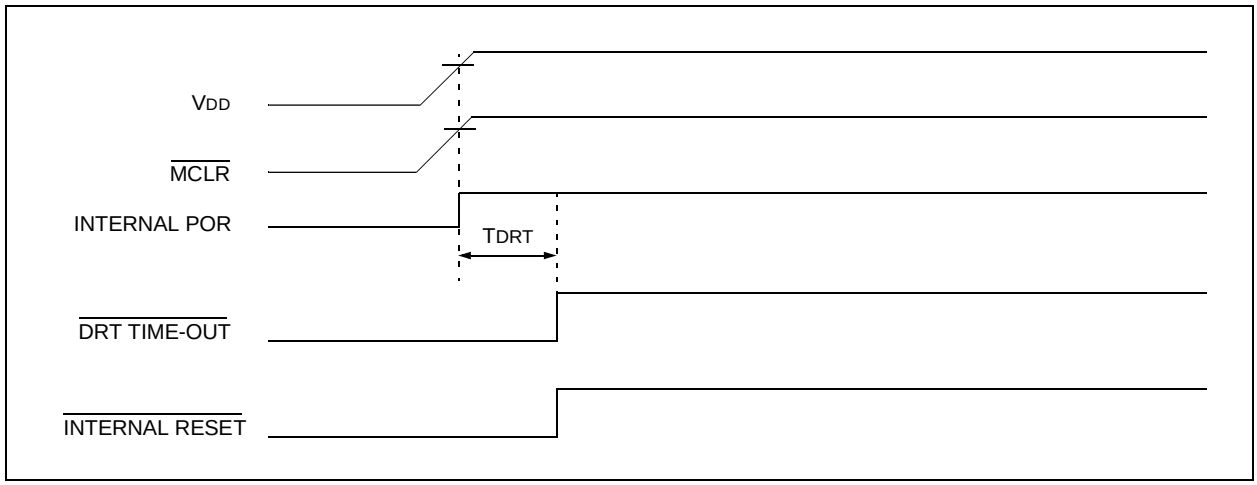
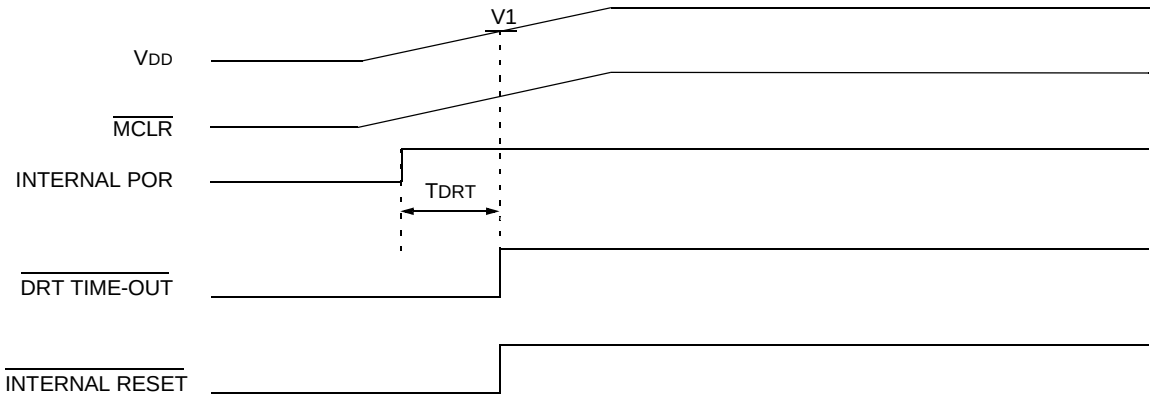


**FIGURE 8-9: TIME-OUT SEQUENCE ON POWER-UP (MCLR TIED TO VDD): SLOW VDD RISE TIME**




When VDD rises slowly, the TDRT time-out expires long before VDD has reached its final value. In this example, the chip will RESET properly if, and only if, V1 ≥ VDD min.

## 8.5 Device Reset Timer (DRT)

In the rfPIC12C509AG/509AF, DRT runs from RESET and varies based on oscillator selection (see Table 8-5).

The DRT operates on an internal RC oscillator. The processor is kept in RESET as long as the DRT is active. The DRT delay allows VDD to rise above VDD min., and for the oscillator to stabilize.

Oscillator circuits based on crystals or ceramic resonators require a certain time after power-up to establish a stable oscillation. The on-chip DRT keeps the device in a RESET condition for approximately 18 ms after MCLR has reached a logic high (VIHMCLR) level. Thus, programming GP3/MCLR/VPP as MCLR and using an external RC network connected to the MCLR input is not required in most cases, allowing for savings in cost-sensitive and/or space restricted applications, as well as allowing the use of the GP3/MCLR/VPP pin as a general purpose input.

The Device Reset time delay will vary from chip to chip due to VDD, temperature, and process variation. See AC parameters for details.

The DRT will also be triggered upon a Watchdog Timer time-out. This is particularly important for applications using the WDT to wake from SLEEP mode automatically.

## 8.6 Watchdog Timer (WDT)

The rfPIC12C509AG/509AF has a Watchdog Timer which can be shut off only through configuration bit WDTE. It runs off of its own RC oscillator for added reliability. If using XT or LP selectable oscillator options, there is always an 18 ms (nominal) delay provided by the Device Reset Timer (DRT), intended to keep the chip in RESET until the crystal oscillator is stable. If using INTRC or EXTRC there is an 18 ms delay only on VDD power-up. With this timer on-chip, most applications need no external RESET circuitry.

The Watchdog Timer (WDT) is a free running on-chip RC oscillator which does not require any external components. This RC oscillator is separate from the external RC oscillator of the GP5/OSC1/CLKIN pin and the internal 4 MHz oscillator. That means that the WDT will run even if the main processor clock has been stopped, for example, by execution of a SLEEP instruction. During normal operation or SLEEP, a WDT Reset or wake-up Reset generates a device RESET.

The $\overline{TO}$ bit (STATUS<4>) will be cleared upon a Watchdog Timer Reset.

The WDT can be permanently disabled by programming the configuration bit WDTE as a '0' (Section 8.1). Refer to the PIC12C5XX Programming Specifications to determine how to access the configuration word.

**TABLE 8-5: DRT (DEVICE RESET TIMER PERIOD)**

| Oscillator Configuration | POR Reset | Subsequent RESETS |
|---|---|---|
| IntRC & ExtRC | 18 ms (typical) | 300 µs (typical) |
| XT & LP | 18 ms (typical) | 18 ms (typical) |

### 8.6.1 WDT PERIOD

The WDT has a nominal time-out period of 18 ms, (with no prescaler). If a longer time-out period is desired, a prescaler with a division ratio of up to 1:128 can be assigned to the WDT (under software control) by writing to the OPTION register. Thus, a time-out period of a nominal 2.3 seconds can be realized. These periods vary with temperature, VDD and part-to-part process variations (see DC specs).

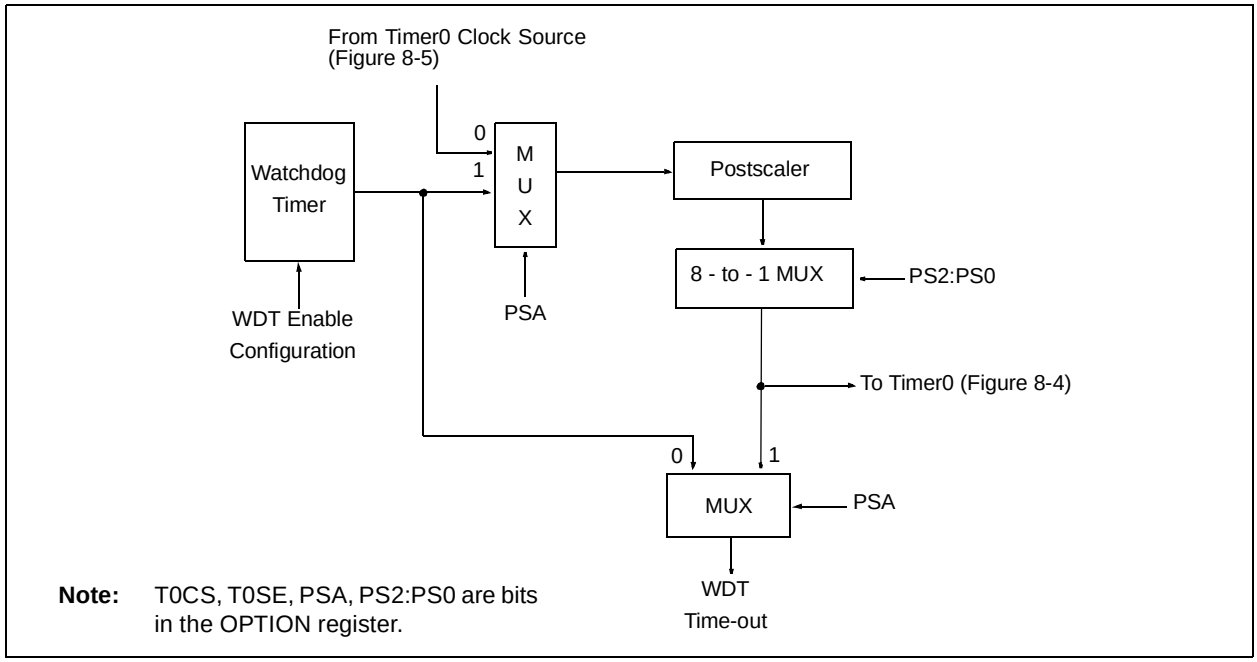
Under worst case conditions (VDD = Min., Temperature = Max., max. WDT prescaler), it may take several seconds before a WDT time-out occurs.

### 8.6.2 WDT PROGRAMMING CONSIDERATIONS

The CLRWDT instruction clears the WDT and the postscaler, if assigned to the WDT, and prevents it from timing out and generating a device RESET.

The SLEEP instruction RESETS the WDT and the postscaler, if assigned to the WDT. This gives the maximum SLEEP time before a WDT wake-up Reset.

**FIGURE 8-10: WATCHDOG TIMER BLOCK DIAGRAM**




**Note:** T0CS, T0SE, PSA, PS2:PS0 are bits in the OPTION register.

**TABLE 8-6: SUMMARY OF REGISTERS ASSOCIATED WITH THE WATCHDOG TIMER**

| Address | Name | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | Value on Power-on Reset | Value on All Other RESETS |
|---|---|---|---|---|---|---|---|---|---|---|---|
| N/A | OPTION | $\overline{GPWU}$ | $\overline{GPPU}$ | T0CS | T0SE | PSA | PS2 | PS1 | PS0 | 1111 1111 | 1111 1111 |

Legend: Shaded boxes = Not used by Watchdog Timer, – = unimplemented, read as '0', u = unchanged

## 8.7 Time-Out Sequence, Power Down, and Wake-up from SLEEP Status Bits (TO/PD/GPWUF)

The $\overline{TO}$, $\overline{PD}$, and GPWUF bits in the STATUS register can be tested to determine if a RESET condition has been caused by a power-up condition, a $\overline{MCLR}$ or Watchdog Timer (WDT) Reset.

**TABLE 8-7: $\overline{TO}$/$\overline{PD}$/GPWUF STATUS AFTER RESET**

| GPWUF | $\overline{TO}$ | $\overline{PD}$ | RESET caused by |
|---|---|---|---|
| 0 | 0 | 0 | WDT wake-up from SLEEP |
| 0 | 0 | u | WDT time-out (not from SLEEP) |
| 0 | 1 | 0 | $\overline{MCLR}$ wake-up from SLEEP |
| 0 | 1 | 1 | Power-up |
| 0 | u | u | $\overline{MCLR}$ not during SLEEP |
| 1 | 1 | 0 | Wake-up from SLEEP on pin change |

Legend: u = unchanged

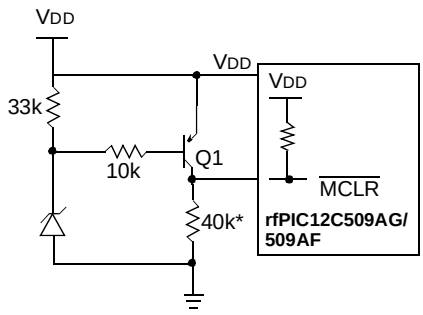
**Note 1:** The $\overline{TO}$, $\overline{PD}$, and GPWUF bits maintain their status (u) until a RESET occurs. A low-pulse on the $\overline{MCLR}$ input does not change the $\overline{TO}$, $\overline{PD}$, and GPWUF status bits.

### 8.7.1 RESET ON BROWN-OUT

A Brown-out is a condition where device power ($V_{DD}$) dips below its minimum value, but not to zero, and then recovers. The device should be RESET in the event of a Brown-out.

To RESET the rfPIC12C509AG/509AF when a Brown-out occurs, external Brown-out protection circuits may be built, as shown in Figure 8-11, Figure 8-12 and Figure 8-13.

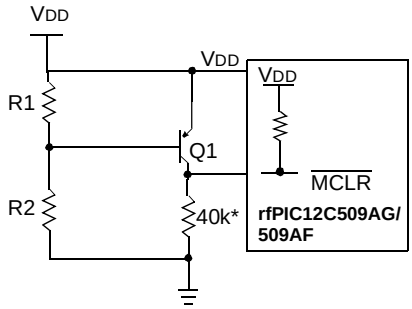
**FIGURE 8-11: BROWN-OUT PROTECTION CIRCUIT 1**




This circuit will activate RESET when $V_{DD}$ goes below Vz + 0.7V (where Vz = Zener voltage).

*Refer to Figure 8-5 and Table 11-1 for internal weak pull-up on $\overline{MCLR}$.

**FIGURE 8-12: BROWN-OUT PROTECTION CIRCUIT 2**

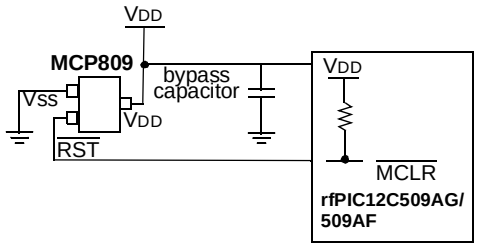



This Brown-out circuit is less expensive, although less accurate. Transistor Q1 turns off when $V_{DD}$ is below a certain level such that:

$$V_{DD} \bullet \frac{R1}{R1 + R2} = 0.7V$$

*Refer to Figure 8-5 and Table 11-1 for internal weak pull-up on $\overline{MCLR}$.

**FIGURE 8-13: BROWN-OUT PROTECTION CIRCUIT 3**




This Brown-out protection circuit employs Microchip Technology's MCP809 microcontroller supervisor. The MCP8XX and MCP1XX family of supervisors provide push-pull and open collector outputs with both high and low active RESET pins. There are 7 different trip point selections to accommodate 5V and 3V systems.

## 8.8 Power-Down Mode (SLEEP)

The SLEEP mode is designed to offer a very low current power-down mode. The user can wake-up from SLEEP through a change on input pins or through a Watchdog Timer time-out. Several oscillator options are also made available to allow the part to fit the application, including an internal 4 MHz oscillator. The EXTRC oscillator option saves system cost while the LP crystal option saves power. A set of configuration bits are used to select various options.

A device may be powered down (SLEEP) and later powered up (Wake-up from SLEEP).

### 8.8.1 SLEEP

The Power-down mode is entered by executing a SLEEP instruction.

If enabled, the Watchdog Timer will be cleared but keeps running, the $\overline{TO}$ bit (STATUS<4>) is set, the $\overline{PD}$ bit (STATUS<3>) is cleared and the oscillator driver is turned off. The I/O ports maintain the status they had before the SLEEP instruction was executed (driving high, driving low, or hi-impedance).

It should be noted that a RESET generated by a WDT time-out does not drive the $\overline{MCLR}$ pin low.

For lowest current consumption while powered down, the T0CKI input should be at $V_{DD}$ or $V_{SS}$ and the GP3/$\overline{MCLR}$/$V_{PP}$ pin must be at a logic high level ($V_{IHMC}$) if $\overline{MCLR}$ is enabled.

### 8.8.2 WAKE-UP FROM SLEEP

The device can wake-up from SLEEP through one of the following events:

1. An external RESET input on GP3/$\overline{MCLR}$/$V_{PP}$ pin, when configured as $\overline{MCLR}$.
2. A Watchdog Timer time-out Reset (if WDT was enabled).
3. A change on input pin GP0, GP1, or GP3/$\overline{MCLR}$/$V_{PP}$ when wake-up on change is enabled.

These events cause a device RESET. The $\overline{TO}$, $\overline{PD}$, and GPWUF bits can be used to determine the cause of device RESET. The $\overline{TO}$ bit is cleared if a WDT time-out occurred (and caused wake-up). The $\overline{PD}$ bit, which is set on power-up, is cleared when SLEEP is invoked.

The GPWUF bit indicates a change in state while in SLEEP at pins GP0, GP1, or GP3 (since the last time there was a file or bit operation on GP port).

| | |
|---|---|
| **Caution:** | Right before entering SLEEP, read the input pins. When in SLEEP, wake up occurs when the values at the pins change from the state they were in at the last reading. If a wake-up on change occurs and the pins are not read before reentering SLEEP, a wake up will occur immediately even if no pins change while in SLEEP mode. |

The WDT is cleared when the device wakes from sleep, regardless of the wake-up source.

## 8.9 Program Verification/Code Protection

If the code protection bit has not been programmed, the on-chip program memory can be read out for verification purposes.

The first 64 locations can be read by the rfPIC12C509AG/509AF regardless of the code protection bit setting.

The last memory location can be read regardless of the code protection bit setting on the rfPIC12C509AG/509AF.

## 8.10 ID Locations

Four memory locations are designated as ID locations where the user can store checksum or other code-identification numbers. These locations are not accessible during normal execution but are readable and writable during program/verify.

Use only the lower 4 bits of the ID locations and always program the upper 8 bits as '0's.

## 8.11 In-Circuit Serial Programming

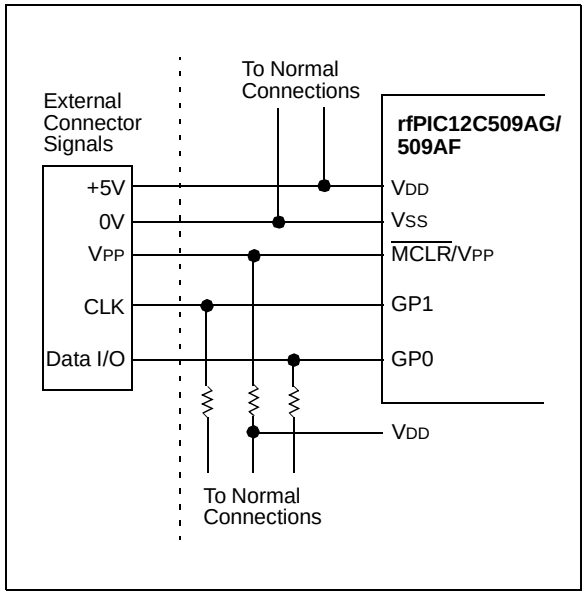
The rfPIC12C509AG/509AF microcontroller with EPROM program memory can be serially programmed while in the end application circuit. This is simply done with two lines for clock and data, and three other lines for power, ground, and the programming voltage. This allows customers to manufacture boards with unprogrammed devices, and then program the microcontroller just before shipping the product. This also allows the most recent firmware or a custom firmware to be programmed.

The device is placed into a program/verify mode by holding the GP1 and GP0 pins low while raising the MCLR ($V_{PP}$) pin from $V_{IL}$ to $V_{IHH}$ (see programming specification). GP1 becomes the programming clock and GP0 becomes the programming data. Both GP1 and GP0 are Schmitt Trigger inputs in this mode.

After RESET, a 6-bit command is then supplied to the device. Depending on the command, 14-bits of program data are then supplied to or from the device, depending if the command was a load or a read. For complete details of serial programming, please refer to the PIC12C5XX Programming Specifications.

A typical in-circuit serial programming connection is shown in Figure 8-14.

**FIGURE 8-14: TYPICAL IN-CIRCUIT SERIAL PROGRAMMING CONNECTION**



**NOTES:**

## 9.0 INSTRUCTION SET SUMMARY

Each rfPIC12C509AG/509AF instruction is a 12-bit word divided into an OPCODE, which specifies the instruction type, and one or more operands which further specify the operation of the instruction. The rfPIC12C509AG/509AF instruction set summary in Table 9-2 groups the instructions into byte-oriented, bit-oriented, and literal and control operations. Table 9-1 shows the opcode field descriptions.

For **byte-oriented** instructions, 'f' represents a file register designator and 'd' represents a destination designator. The file register designator is used to specify which one of the 32 file registers is to be used by the instruction.

The destination designator specifies where the result of the operation is to be placed. If 'd' is '0', the result is placed in the W register. If 'd' is '1', the result is placed in the file register specified in the instruction.

For **bit-oriented** instructions, 'b' represents a bit field designator which selects the number of the bit affected by the operation, while 'f' represents the number of the file in which the bit is located.

For **literal and control** operations, 'k' represents an 8 or 9-bit constant or literal value.

**TABLE 9-1: OPCODE FIELD DESCRIPTIONS**

| Field | Description |
|---|---|
| f | Register file address (0x00 to 0x7F) |
| W | Working register (accumulator) |
| b | Bit address within an 8-bit file register |
| k | Literal field, constant data or label |
| x | Don't care location (= 0 or 1)<br>The assembler will generate code with x = 0. It is the recommended form of use for compatibility with all Microchip software tools. |
| d | Destination select;<br>d = 0 (store result in W)<br>d = 1 (store result in file register 'f')<br>Default is d = 1 |
| label | Label name |
| TOS | Top of Stack |
| PC | Program Counter |
| WDT | Watchdog Timer Counter |
| $\overline{TO}$ | Time-Out bit |
| $\overline{PD}$ | Power-Down bit |
| dest | Destination, either the W register or the specified register file location |
| [ ] | Options |
| ( ) | Contents |
| → | Assigned to |
| < > | Register bit field |
| ∈ | In the set of |
| *italics* | User defined term (font is courier) |

All instructions are executed within a single instruction cycle, unless a conditional test is true or the program counter is changed as a result of an instruction. In this case, the execution takes two instruction cycles. One instruction cycle consists of four oscillator periods. Thus, for an oscillator frequency of 4 MHz, the normal instruction execution time is 1 µs. If a conditional test is true or the program counter is changed as a result of an instruction, the instruction execution time is 2 µs.

Figure 9-1 shows the three general formats that the instructions can have. All examples in the figure use the following format to represent a hexadecimal number:

0xhhh

where 'h' signifies a hexadecimal digit.

**FIGURE 9-1: GENERAL FORMAT FOR INSTRUCTIONS**

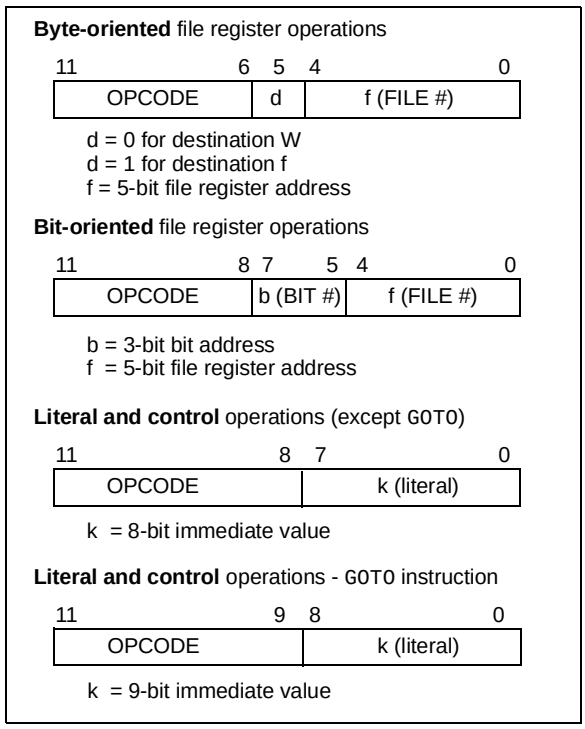


**TABLE 9-2: INSTRUCTION SET SUMMARY**

| Mnemonic, Operands | | Description | Cycles | 12-Bit Opcode MSb | LSb | Status Affected | Notes |
|---|---|---|---|---|---|---|---|
| **ADDWF** | **f,d** | Add W and f | 1 | 0001 11df ffff | | C,DC,Z | **1,2,4** |
| **ANDWF** | **f,d** | AND W with f | 1 | 0001 01df ffff | | Z | **2,4** |
| **CLRF** | **f** | Clear f | 1 | 0000 011f ffff | | Z | **4** |
| **CLRW** | **–** | Clear W | 1 | 0000 0100 0000 | | Z | |
| **COMF** | **f, d** | Complement f | 1 | 0010 01df ffff | | Z | |
| **DECF** | **f, d** | Decrement f | 1 | 0000 11df ffff | | Z | **2,4** |
| **DECFSZ** | **f, d** | Decrement f, Skip if 0 | 1(2) | 0010 11df ffff | | None | **2,4** |
| **INCF** | **f, d** | Increment f | 1 | 0010 10df ffff | | Z | **2,4** |
| **INCFSZ** | **f, d** | Increment f, Skip if 0 | 1(2) | 0011 11df ffff | | None | **2,4** |
| **IORWF** | **f, d** | Inclusive OR W with f | 1 | 0001 00df ffff | | Z | **2,4** |
| **MOVF** | **f, d** | Move f | 1 | 0010 00df ffff | | Z | **2,4** |
| **MOVWF** | **f** | Move W to f | 1 | 0000 001f ffff | | None | **1,4** |
| **NOP** | **–** | No Operation | 1 | 0000 0000 0000 | | None | |
| **RLF** | **f, d** | Rotate left f through Carry | 1 | 0011 01df ffff | | C | **2,4** |
| **RRF** | **f, d** | Rotate right f through Carry | 1 | 0011 00df ffff | | C | **2,4** |
| **SUBWF** | **f, d** | Subtract W from f | 1 | 0000 10df ffff | | C,DC,Z | **1,2,4** |
| **SWAPF** | **f, d** | Swap f | 1 | 0011 10df ffff | | None | **2,4** |
| **XORWF** | **f, d** | Exclusive OR W with f | 1 | 0001 10df ffff | | Z | **2,4** |
| **BIT-ORIENTED FILE REGISTER OPERATIONS** | | | | | | | |
| **BCF** | **f, b** | Bit Clear f | 1 | 0100 bbbf ffff | | None | **2,4** |
| **BSF** | **f, b** | Bit Set f | 1 | 0101 bbbf ffff | | None | **2,4** |
| **BTFSC** | **f, b** | Bit Test f, Skip if Clear | 1 (2) | 0110 bbbf ffff | | None | |
| **BTFSS** | **f, b** | Bit Test f, Skip if Set | 1 (2) | 0111 bbbf ffff | | None | |
| **LITERAL AND CONTROL OPERATIONS** | | | | | | | |
| **ANDLW** | **k** | AND literal with W | 1 | 1110 kkkk kkkk | | Z | |
| **CALL** | **k** | Call subroutine | 2 | 1001 kkkk kkkk | | None | **1** |
| **CLRWDT** | **k** | Clear Watchdog Timer | 1 | 0000 0000 0100 | | $\overline{TO}$, $\overline{PD}$ | |
| **GOTO** | **k** | Unconditional branch | 2 | 101k kkkk kkkk | | None | |
| **IORLW** | **k** | Inclusive OR Literal with W | 1 | 1101 kkkk kkkk | | Z | |
| **MOVLW** | **k** | Move Literal to W | 1 | 1100 kkkk kkkk | | None | |
| **OPTION** | **–** | Load OPTION register | 1 | 0000 0000 0010 | | None | |
| **RETLW** | **k** | Return, place Literal in W | 2 | 1000 kkkk kkkk | | None | |
| **SLEEP** | **–** | Go into standby mode | 1 | 0000 0000 0011 | | $\overline{TO}$, $\overline{PD}$ | |
| **TRIS** | **f** | Load TRIS register | 1 | 0000 0000 0fff | | None | **3** |
| **XORLW** | **k** | Exclusive OR Literal to W | 1 | 1111 kkkk kkkk | | Z | |

**Note 1:** The 9th bit of the program counter will be forced to a '0' by any instruction that writes to the PC except for GOTO. (Section 4.6)

**2:** When an I/O register is modified as a function of itself (e.g. MOVF GPIO, 1), the value used will be that value present on the pins themselves. For example, if the data latch is '1' for a pin configured as input and is driven low by an external device, the data will be written back with a '0'.

**3:** The instruction TRIS f, where f = 6 causes the contents of the W register to be written to the tristate latches of GPIO. A '1' forces the pin to a hi-impedance state and disables the output buffers.

**4:** If this instruction is executed on the TMR0 register (and, where applicable, d = 1), the prescaler will be cleared (if assigned to TMR0).

**ADDWF** **Add W and f**

Syntax: [ *label* ] ADDWF f,d

Operands: $0 \le f \le 31$
$d \in [0,1]$

Operation: (W) + (f) → (dest)

Status Affected: C, DC, Z

Encoding:

| | | |
|---|---|---|
| 0001 | 11df | ffff |

Description: Add the contents of the W register and register 'f'. If 'd' is 0 the result is stored in the W register. If 'd' is '1' the result is stored back in register 'f'.

Words: 1

Cycles: 1

Example:
```
ADDWF   FSR, 0
```
Before Instruction
W = 0x17
FSR = 0xC2

After Instruction
W = 0xD9
FSR = 0xC2

**ANDWF** **AND W with f**

Syntax: [ *label* ] ANDWF f,d

Operands: $0 \le f \le 31$
$d \in [0,1]$

Operation: (W) .AND. (f) → (dest)

Status Affected: Z

Encoding:

| | | |
|---|---|---|
| 0001 | 01df | ffff |

Description: The contents of the W register are AND'ed with register 'f'. If 'd' is 0 the result is stored in the W register. If 'd' is '1' the result is stored back in register 'f'.

Words: 1

Cycles: 1

Example:
```
ANDWF   FSR,    1
```
Before Instruction
W = 0x17
FSR = 0xC2

After Instruction
W = 0x17
FSR = 0x02

**ANDLW** **And literal with W**

Syntax: [ *label* ] ANDLW k

Operands: $0 \le k \le 255$

Operation: (W).AND. (k) → (W)

Status Affected: Z

Encoding:

| | | |
|---|---|---|
| 1110 | kkkk | kkkk |

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

Words: 1

Cycles: 1

Example:
```
ANDLW   0x5F
```
Before Instruction
W = 0xA3

After Instruction
W = 0x03

**BCF** **Bit Clear f**

Syntax: [ *label* ] BCF f,b

Operands: $0 \le f \le 31$
$0 \le b \le 7$

Operation: 0 → (f<b>)

Status Affected: None

Encoding:

| | | |
|---|---|---|
| 0100 | bbbf | ffff |

Description: Bit 'b' in register 'f' is cleared.

Words: 1

Cycles: 1

Example:
```
BCF     FLAG_REG,  7
```
Before Instruction
FLAG_REG = 0xC7

After Instruction
FLAG_REG = 0x47

**BSF** **Bit Set f**

Syntax: [ *label* ] BSF f,b

Operands: 0 ≤ f ≤ 31
0 ≤ b ≤ 7

Operation: 1 → (f<b>)

Status Affected: None

Encoding:

| 0101 | bbbf | ffff |
|---|---|---|

Description: Bit 'b' in register 'f' is set.

Words: 1

Cycles: 1

Example:
```
BSF     FLAG_REG,   7
```
Before Instruction
FLAG_REG = 0x0A
After Instruction
FLAG_REG = 0x8A

**BTFSC** **Bit Test f, Skip if Clear**

Syntax: [ *label* ] BTFSC f,b

Operands: 0 ≤ f ≤ 31
0 ≤ b ≤ 7

Operation: skip if (f<b>) = 0

Status Affected: None

Encoding:

| 0110 | bbbf | ffff |
|---|---|---|

Description: If bit 'b' in register 'f' is 0 then the next instruction is skipped.
If bit 'b' is 0 then the next instruction fetched during the current instruction execution is discarded, and an NOP is executed instead, making this a 2 cycle instruction.

Words: 1

Cycles: 1(2)

Example:
```
HERE    BTFSC   FLAG,1
FALSE   GOTO    PROCESS_CODE
TRUE    •
        •
        •
```
Before Instruction
PC = address (HERE)
After Instruction
if FLAG<1> = 0,
PC = address (TRUE);
if FLAG<1> = 1,
PC = address(FALSE)

**BTFSS** **Bit Test f, Skip if Set**

Syntax: [ *label* ] BTFSS f,b

Operands: 0 ≤ f ≤ 31
0 ≤ b < 7

Operation: skip if (f<b>) = 1

Status Affected: None

Encoding:

| 0111 | bbbf | ffff |
|---|---|---|

Description: If bit 'b' in register 'f' is '1' then the next instruction is skipped.
If bit 'b' is '1', then the next instruction fetched during the current instruction execution, is discarded and an NOP is executed instead, making this a 2 cycle instruction.

Words: 1

Cycles: 1(2)

Example:
```
HERE    BTFSS   FLAG,1
FALSE   GOTO    PROCESS_CODE
TRUE    •
        •
        •
```
Before Instruction
PC = address (HERE)
After Instruction
If FLAG<1> = 0,
PC = address (FALSE);
if FLAG<1> = 1,
PC = address (TRUE)

**CALL** **Subroutine Call**

Syntax: [ *label* ] CALL k

Operands: 0 ≤ k ≤ 255

Operation: (PC) + 1→ Top of Stack;
k → PC<7:0>;
(STATUS<6:5>) → PC<10:9>;
0 → PC<8>

Status Affected: None

Encoding:

| | | |
|---|---|---|
| 1001 | kkkk | kkkk |

Description: Subroutine call. First, return address (PC+1) is pushed onto the stack. The eight bit immediate address is loaded into PC bits <7:0>. The upper bits PC<10:9> are loaded from STATUS<6:5>, PC<8> is cleared. CALL is a two cycle instruction.

Words: 1

Cycles: 2

Example:
```
HERE    CALL    THERE
```
Before Instruction
PC = address (HERE)
After Instruction
PC = address (THERE)
TOS = address (HERE + 1)

**CLRF** **Clear f**

Syntax: [ *label* ] CLRF f

Operands: 0 ≤ f ≤ 31

Operation: 00h → (f);
1 → Z

Status Affected: Z

Encoding:

| | | |
|---|---|---|
| 0000 | 011f | ffff |

Description: The contents of register 'f' are cleared and the Z bit is set.

Words: 1

Cycles: 1

Example:
```
CLRF    FLAG_REG
```
Before Instruction
FLAG_REG = 0x5A
After Instruction
FLAG_REG = 0x00
Z = 1

**CLRW** **Clear W**

Syntax: [ *label* ] CLRW

Operands: None

Operation: 00h → (W);
1 → Z

Status Affected: Z

Encoding:

| | | |
|---|---|---|
| 0000 | 0100 | 0000 |

Description: The W register is cleared. Zero bit (Z) is set.

Words: 1

Cycles: 1

Example:
```
CLRW
```
Before Instruction
W = 0x5A
After Instruction
W = 0x00
Z = 1

**CLRWDT** **Clear Watchdog Timer**

Syntax: [ *label* ] CLRWDT

Operands: None

Operation: 00h → WDT;
0 → $\overline{\text{WDT}}$ prescaler (if assigned);
1 → $\overline{\text{TO}}$;
1 → $\overline{\text{PD}}$

Status Affected: $\overline{\text{TO}}$, $\overline{\text{PD}}$

Encoding:

| | | |
|---|---|---|
| 0000 | 0000 | 0100 |

Description: The CLRWDT instruction RESETS the WDT. It also RESETS the prescaler, if the prescaler is assigned to the $\overline{\text{WDT}}$ and not Timer0. Status bits $\overline{\text{TO}}$ and $\overline{\text{PD}}$ are set.

Words: 1

Cycles: 1

Example:
```
CLRWDT
```
Before Instruction
WDT counter = ?
After Instruction
WDT counter = 0x00
$\overline{\text{WDT}}$ prescale = 0
$\overline{\text{TO}}$ = 1
$\overline{\text{PD}}$ = 1

**COMF** **Complement f**

Syntax: [ *label* ] COMF f,d

Operands: $0 \le f \le 31$
$d \in [0,1]$

Operation: $(\bar{f}) \rightarrow (dest)$

Status Affected: Z

Encoding:

| 0010 | 01df | ffff |
|---|---|---|

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

Words: 1

Cycles: 1

Example:
```
COMF    REG1,0
```
Before Instruction
REG1 = 0x13

After Instruction
REG1 = 0x13
W = 0xEC

**DECF** **Decrement f**

Syntax: [ *label* ] DECF f,d

Operands: $0 \le f \le 31$
$d \in [0,1]$

Operation: $(f) - 1 \rightarrow (dest)$

Status Affected: Z

Encoding:

| 0000 | 11df | ffff |
|---|---|---|

Description: Decrement register 'f'. If 'd' is 0 the result is stored in the W register. If 'd' is 1 the result is stored back in register 'f'.

Words: 1

Cycles: 1

Example:
```
DECF    CNT,    1
```
Before Instruction
CNT = 0x01
Z = 0

After Instruction
CNT = 0x00
Z = 1

**DECFSZ** **Decrement f, Skip if 0**

Syntax: [ *label* ] DECFSZ f,d

Operands: $0 \le f \le 31$
$d \in [0,1]$

Operation: $(f) - 1 \rightarrow d$; skip if result = 0

Status Affected: None

Encoding:

| 0010 | 11df | ffff |
|---|---|---|

Description: The contents of register 'f' are decremented. If 'd' is 0 the result is placed in the W register. If 'd' is 1 the result is placed back in register 'f'.
If the result is 0, the next instruction, which is already fetched, is discarded and an NOP is executed instead making it a two cycle instruction.

Words: 1

Cycles: 1(2)

Example:
```
HERE      DECFSZ   CNT, 1
          GOTO     LOOP
CONTINUE  •
          •
          •
```
Before Instruction
PC = address (HERE)

After Instruction
CNT = CNT - 1;
if CNT = 0,
PC = address (CONTINUE);
if CNT ≠ 0,
PC = address (HERE+1)

**GOTO** **Unconditional Branch**

Syntax: [ *label* ] GOTO k

Operands: $0 \le k \le 511$

Operation: $k \rightarrow PC<8:0>$;
$STATUS<6:5> \rightarrow PC<10:9>$

Status Affected: None

Encoding:

| 101k | kkkk | kkkk |
|---|---|---|

Description: GOTO is an unconditional branch. The 9-bit immediate value is loaded into PC bits <8:0>. The upper bits of PC are loaded from STATUS<6:5>. GOTO is a two cycle instruction.

Words: 1

Cycles: 2

Example:
```
GOTO THERE
```
After Instruction
PC = address (THERE)

**INCF** **Increment f**

Syntax: [ *label* ] INCF f,d

Operands: $0 \le f \le 31$
$d \in [0,1]$

Operation: (f) + 1 → (dest)

Status Affected: Z

Encoding:

| 0010 | 10df | ffff |
|---|---|---|

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

Words: 1

Cycles: 1

Example:
```
INCF    CNT,    1
```
Before Instruction
CNT = 0xFF
Z = 0

After Instruction
CNT = 0x00
Z = 1

**INCFSZ** **Increment f, Skip if 0**

Syntax: [ *label* ] INCFSZ f,d

Operands: $0 \le f \le 31$
$d \in [0,1]$

Operation: (f) + 1 → (dest), skip if result = 0

Status Affected: None

Encoding:

| 0011 | 11df | ffff |
|---|---|---|

Description: The contents of register 'f' are incremented. If 'd' is 0 the result is placed in the W register. If 'd' is 1 the result is placed back in register 'f'.
If the result is 0, then the next instruction, which is already fetched, is discarded and an NOP is executed instead making it a two cycle instruction.

Words: 1

Cycles: 1(2)

Example:
```
HERE    INCFSZ    CNT,  1
        GOTO      LOOP
CONTINUE •
         •
         •
```
Before Instruction
PC = address (HERE)

After Instruction
CNT = CNT + 1;
if CNT = 0,
PC = address (CONTINUE);
if CNT ≠ 0,
PC = address (HERE +1)

**IORLW** **Inclusive OR literal with W**

Syntax: [ *label* ] IORLW k

Operands: $0 \le k \le 255$

Operation: (W) .OR. (k) → (W)

Status Affected: Z

Encoding:

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

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

Words: 1

Cycles: 1

Example:
```
IORLW   0x35
```
Before Instruction
W = 0x9A

After Instruction
W = 0xBF
Z = 0

**IORWF** **Inclusive OR W with f**

Syntax: [ *label* ] IORWF f,d

Operands: $0 \le f \le 31$
$d \in [0,1]$

Operation: (W).OR. (f) → (dest)

Status Affected: Z

Encoding:

| 0001 | 00df | ffff |
|---|---|---|

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

Words: 1

Cycles: 1

Example:
```
IORWF           RESULT, 0
```
Before Instruction
RESULT = 0x13
W = 0x91

After Instruction
RESULT = 0x13
W = 0x93
Z = 0

**MOVF** **Move f**

Syntax: [ *label* ] MOVF f,d

Operands: $0 \le f \le 31$
$d \in [0,1]$

Operation: (f) → (dest)

Status Affected: Z

Encoding:

| | | |
|---|---|---|
| 0010 | 00df | ffff |

Description: The contents of register 'f' is moved to destination 'd'. If 'd' is 0, destination is the W register. If 'd' is 1, the destination is file register 'f'. 'd' is 1 is useful to test a file register since status flag Z is affected.

Words: 1

Cycles: 1

Example:
```
MOVF    FSR,    0
```
After Instruction
W = value in FSR register

**MOVLW** **Move Literal to W**

Syntax: [ *label* ] MOVLW k

Operands: $0 \le k \le 255$

Operation: k → (W)

Status Affected: None

Encoding:

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

Description: The eight bit literal 'k' is loaded into the W register. The don't cares will assemble as 0s.

Words: 1

Cycles: 1

Example:
```
MOVLW   0x5A
```
After Instruction
W = 0x5A

**MOVWF** **Move W to f**

Syntax: [ *label* ] MOVWF f

Operands: $0 \le f \le 31$

Operation: (W) → (f)

Status Affected: None

Encoding:

| | | |
|---|---|---|
| 0000 | 001f | ffff |

Description: Move data from the W register to register 'f'.

Words: 1

Cycles: 1

Example:
```
MOVWF   TEMP_REG
```
Before Instruction
TEMP_REG = 0xFF
W = 0x4F

After Instruction
TEMP_REG = 0x4F
W = 0x4F

**NOP** **No Operation**

Syntax: [ *label* ] NOP

Operands: None

Operation: No operation

Status Affected: None

Encoding:

| | | |
|---|---|---|
| 0000 | 0000 | 0000 |

Description: No operation.

Words: 1

Cycles: 1

Example:
```
NOP
```

**OPTION** **Load OPTION Register**

Syntax: [ *label* ] OPTION

Operands: None

Operation: (W) → OPTION

Status Affected: None

Encoding:

| | | |
|---|---|---|
| 0000 | 0000 | 0010 |

Description: The content of the W register is loaded into the OPTION register.

Words: 1

Cycles: 1

Example
```
OPTION
```
Before Instruction
W = 0x07

After Instruction
OPTION = 0x07

**RETLW** **Return with Literal in W**

Syntax: [ *label* ] RETLW k

Operands: 0 ≤ k ≤ 255

Operation: k → (W);
TOS → PC

Status Affected: None

Encoding:

| | | |
|---|---|---|
| 1000 | kkkk | kkkk |

Description: The W register is loaded with the eight bit literal 'k'. The program counter is loaded from the top of the stack (the return address). This is a two cycle instruction.

Words: 1

Cycles: 2

Example:
```
CALL TABLE ;W contains
           ;table offset
           ;value.
•          ;W now has table
•          ;value.
•
TABLE      ADDWF PC   ;W = offset
           RETLW k1   ;Begin table
           RETLW k2   ;
           •
           •
           •
           RETLW kn   ; End of table
```
Before Instruction
W = 0x07

After Instruction
W = value of k8

**RLF** **Rotate Left f through Carry**

Syntax: [ *label* ] RLF f,d

Operands: 0 ≤ f ≤ 31
d ∈ [0,1]

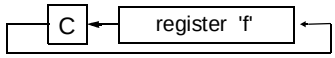
Operation: See description below

Status Affected: C

Encoding:

| | | |
|---|---|---|
| 0011 | 01df | ffff |

Description: The contents of register 'f' are rotated one bit to the left through the Carry Flag. If 'd' is 0 the result is placed in the W register. If 'd' is 1 the result is stored back in register 'f'.




Words: 1

Cycles: 1

Example:
```
RLF     REG1,0
```
Before Instruction
```
REG1    =   1110 0110
C       =   0
```
After Instruction
```
REG1    =   1110 0110
W       =   1100 1100
C       =   1
```

**RRF** **Rotate Right f through Carry**

Syntax: [ *label* ] RRF f,d

Operands: 0 ≤ f ≤ 31
d ∈ [0,1]

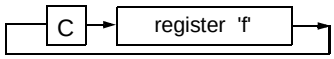
Operation: See description below

Status Affected: C

Encoding:

| | | |
|---|---|---|
| 0011 | 00df | ffff |

Description: The contents of register 'f' are rotated one bit to the right through the Carry Flag. If 'd' is 0 the result is placed in the W register. If 'd' is 1 the result is placed back in register 'f'.




Words: 1

Cycles: 1

Example:
```
RRF     REG1,0
```
Before Instruction
```
REG1    =   1110 0110
C       =   0
```
After Instruction
```
REG1    =   1110 0110
W       =   0111 0011
C       =   0
```

**SLEEP** **Enter SLEEP Mode**

Syntax: [*label*] SLEEP

Operands: None

Operation: 00h → WDT;
0 → WDT prescaler;
1 → $\overline{TO}$;
0 → $\overline{PD}$

Status Affected: $\overline{TO}$, $\overline{PD}$, GPWUF

Encoding:

| | | |
|---|---|---|
| 0000 | 0000 | 0011 |

Description: Time-out status bit ($\overline{TO}$) is set. The power down status bit ($\overline{PD}$) is cleared.
GPWUF is unaffected.
The WDT and its prescaler are cleared.
The processor is put into SLEEP mode with the oscillator stopped. See section on SLEEP for more details.

Words: 1

Cycles: 1

Example: SLEEP

**SUBWF** **Subtract W from f**

Syntax: [*label*] SUBWF f,d

Operands: $0 \le f \le 31$
$d \in [0,1]$

Operation: (f) – (W) → (dest)

Status Affected: C, DC, Z

Encoding:

| | | |
|---|---|---|
| 0000 | 10df | ffff |

Description: Subtract (2's complement method) the W register from register 'f'. If 'd' is 0 the result is stored in the W register. If 'd' is 1 the result is stored back in register 'f'.

Words: 1

Cycles: 1

Example 1:

```
SUBWF   REG1, 1
```

Before Instruction
REG1 = 3
W = 2
C = ?

After Instruction
REG1 = 1
W = 2
C = 1 ; result is positive

Example 2:

Before Instruction
REG1 = 2
W = 2
C = ?

After Instruction
REG1 = 0
W = 2
C = 1 ; result is zero

Example 3:

Before Instruction
REG1 = 1
W = 2
C = ?

After Instruction
REG1 = FF
W = 2
C = 0 ; result is negative

**SWAPF** **Swap Nibbles in f**

Syntax: [ *label* ] SWAPF f,d

Operands: $0 \le f \le 31$
$d \in [0,1]$

Operation: (f<3:0>) → (dest<7:4>);
(f<7:4>) → (dest<3:0>)

Status Affected: None

Encoding:

| 0011 | 10df | ffff |
|---|---|---|

Description: The upper and lower nibbles of register 'f' are exchanged. If 'd' is 0 the result is placed in W register. If 'd' is 1 the result is placed in register 'f'.

Words: 1

Cycles: 1

Example
```
SWAPF   REG1,  0
```
Before Instruction
REG1 = 0xA5
After Instruction
REG1 = 0xA5
W = 0X5A

**TRIS** **Load TRIS Register**

Syntax: [ *label* ] TRIS f

Operands: f = 6

Operation: (W) → TRIS register f

Status Affected: None

Encoding:

| 0000 | 0000 | 0fff |
|---|---|---|

Description: TRIS register 'f' (f = 6) is loaded with the contents of the W register

Words: 1

Cycles: 1

Example
```
TRIS    GPIO
```
Before Instruction
W = 0XA5
After Instruction
TRIS = 0XA5

**Note:** f = 6 for PIC12C5XX only.

**XORLW** **Exclusive OR literal with W**

Syntax: [*label*] XORLW k

Operands: $0 \le k \le 255$

Operation: (W) .XOR. k → (W)

Status Affected: Z

Encoding:

| 1111 | kkkk | kkkk |
|---|---|---|

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

Words: 1

Cycles: 1

Example
```
XORLW  0xAF
```
Before Instruction
W = 0xB5
After Instruction
W = 0x1A

**XORWF** **Exclusive OR W with f**

Syntax: [ *label* ] XORWF f,d

Operands: $0 \le f \le 31$
$d \in [0,1]$

Operation: (W) .XOR. (f) → (dest)

Status Affected: Z

Encoding:

| 0001 | 10df | ffff |
|---|---|---|

Description: Exclusive OR the contents of the W register with register 'f'. If 'd' is 0 the result is stored in the W register. If 'd' is 1 the result is stored back in register 'f'.

Words: 1

Cycles: 1

Example
```
XORWF  REG,1
```
Before Instruction
REG = 0xAF
W = 0xB5
After Instruction
REG = 0x1A
W = 0xB5

**NOTES:**

## 10.0 DEVELOPMENT SUPPORT

The PICmicro® microcontrollers are supported with a full range of hardware and software development tools:

- Integrated Development Environment
  - MPLAB® IDE Software
- Assemblers/Compilers/Linkers
  - MPASM™ Assembler
  - MPLAB C17 and MPLAB C18 C Compilers
  - MPLINK™ Object Linker/ MPLIB™ Object Librarian
- Simulators
  - MPLAB SIM Software Simulator
- Emulators
  - MPLAB ICE 2000 In-Circuit Emulator
  - ICEPIC™ In-Circuit Emulator
- In-Circuit Debugger
  - MPLAB ICD
- Device Programmers
  - PRO MATE® II Universal Device Programmer
  - PICSTART® Plus Entry-Level Development Programmer
- Low Cost Demonstration Boards
  - PICDEM™ 1 Demonstration Board
  - PICDEM 2 Demonstration Board
  - PICDEM 3 Demonstration Board
  - PICDEM 17 Demonstration Board
  - KEELOQ® Demonstration Board

### 10.1 MPLAB Integrated Development Environment Software

The MPLAB IDE software brings an ease of software development previously unseen in the 8-bit microcontroller market. The MPLAB IDE is a Windows®-based application that contains:

- An interface to debugging tools
  - simulator
  - programmer (sold separately)
  - emulator (sold separately)
  - in-circuit debugger (sold separately)
- A full-featured editor
- A project manager
- Customizable toolbar and key mapping
- A status bar
- On-line help

The MPLAB IDE allows you to:

- Edit your source files (either assembly or ‘C’)
- One touch assemble (or compile) and download to PICmicro emulator and simulator tools (automatically updates all project information)
- Debug using:
  - source files
  - absolute listing file
  - machine code

The ability to use MPLAB IDE with multiple debugging tools allows users to easily switch from the cost-effective simulator to a full-featured emulator with minimal retraining.

### 10.2 MPASM Assembler

The MPASM assembler is a full-featured universal macro assembler for all PICmicro MCU’s.

The MPASM assembler has a command line interface and a Windows shell. It can be used as a stand-alone application on a Windows 3.x or greater system, or it can be used through MPLAB IDE. The MPASM assembler generates relocatable object files for the MPLINK object linker, Intel® standard HEX files, MAP files to detail memory usage and symbol reference, an absolute LST file that contains source lines and generated machine code, and a COD file for debugging.

The MPASM assembler features include:

- Integration into MPLAB IDE projects.
- User-defined macros to streamline assembly code.
- Conditional assembly for multi-purpose source files.
- Directives that allow complete control over the assembly process.

### 10.3 MPLAB C17 and MPLAB C18 C Compilers

The MPLAB C17 and MPLAB C18 Code Development Systems are complete ANSI ‘C’ compilers for Microchip’s PIC17CXXX and PIC18CXXX family of microcontrollers, respectively. These compilers provide powerful integration capabilities and ease of use not found with other compilers.

For easier source level debugging, the compilers provide symbol information that is compatible with the MPLAB IDE memory display.

## 10.4 MPLINK Object Linker/ MPLIB Object Librarian

The MPLINK object linker combines relocatable objects created by the MPASM assembler and the MPLAB C17 and MPLAB C18 C compilers. It can also link relocatable objects from pre-compiled libraries, using directives from a linker script.

The MPLIB object librarian is a librarian for pre-compiled code to be used with the MPLINK object linker. When a routine from a library is called from another source file, only the modules that contain that routine will be linked in with the application. This allows large libraries to be used efficiently in many different applications. The MPLIB object librarian manages the creation and modification of library files.

The MPLINK object linker features include:

- Integration with MPASM assembler and MPLAB C17 and MPLAB C18 C compilers.
- Allows all memory areas to be defined as sections to provide link-time flexibility.

The MPLIB object librarian features include:

- Easier linking because single libraries can be included instead of many smaller files.
- Helps keep code maintainable by grouping related modules together.
- Allows libraries to be created and modules to be added, listed, replaced, deleted or extracted.

## 10.5 MPLAB SIM Software Simulator

The MPLAB SIM software simulator allows code development in a PC-hosted environment by simulating the PICmicro series microcontrollers on an instruction level. On any given instruction, the data areas can be examined or modified and stimuli can be applied from a file, or user-defined key press, to any of the pins. The execution can be performed in single step, execute until break, or trace mode.

The MPLAB SIM simulator fully supports symbolic debugging using the MPLAB C17 and the MPLAB C18 C compilers and the MPASM assembler. The software simulator offers the flexibility to develop and debug code outside of the laboratory environment, making it an excellent multi-project software development tool.

## 10.6 MPLAB ICE High Performance Universal In-Circuit Emulator with MPLAB IDE

The MPLAB ICE universal in-circuit emulator is intended to provide the product development engineer with a complete microcontroller design tool set for PICmicro microcontrollers (MCUs). Software control of the MPLAB ICE in-circuit emulator is provided by the MPLAB Integrated Development Environment (IDE), which allows editing, building, downloading and source debugging from a single environment.

The MPLAB ICE 2000 is a full-featured emulator system with enhanced trace, trigger and data monitoring features. Interchangeable processor modules allow the system to be easily reconfigured for emulation of different processors. The universal architecture of the MPLAB ICE in-circuit emulator allows expansion to support new PICmicro microcontrollers.

The MPLAB ICE in-circuit emulator system has been designed as a real-time emulation system, with advanced features that are generally found on more expensive development tools. The PC platform and Microsoft® Windows environment were chosen to best make these features available to you, the end user.

## 10.7 ICEPIC In-Circuit Emulator

The ICEPIC low cost, in-circuit emulator is a solution for the Microchip Technology PIC16C5X, PIC16C6X, PIC16C7X and PIC16CXXX families of 8-bit One-Time-Programmable (OTP) microcontrollers. The modular system can support different subsets of PIC16C5X or PIC16CXXX products through the use of interchangeable personality modules, or daughter boards. The emulator is capable of emulating without target application circuitry being present.

## 10.8 MPLAB ICD In-Circuit Debugger

Microchip's In-Circuit Debugger, MPLAB ICD, is a powerful, low cost, run-time development tool. This tool is based on the FLASH PICmicro MCUs and can be used to develop for this and other PICmicro microcontrollers. The MPLAB ICD utilizes the in-circuit debugging capability built into the FLASH devices. This feature, along with Microchip's In-Circuit Serial Programming™ protocol, offers cost-effective in-circuit FLASH debugging from the graphical user interface of the MPLAB Integrated Development Environment. This enables a designer to develop and debug source code by watching variables, single-stepping and setting break points. Running at full speed enables testing hardware in real-time.

## 10.9 PRO MATE II Universal Device Programmer

The PRO MATE II universal device programmer is a full-featured programmer, capable of operating in stand-alone mode, as well as PC-hosted mode. The PRO MATE II device programmer is CE compliant.

The PRO MATE II device programmer has programmable $V_{DD}$ and $V_{PP}$ supplies, which allow it to verify programmed memory at $V_{DD}$ min and $V_{DD}$ max for maximum reliability. It has an LCD display for instructions and error messages, keys to enter commands and a modular detachable socket assembly to support various package types. In stand-alone mode, the PRO MATE II device programmer can read, verify, or program PICmicro devices. It can also set code protection in this mode.

## 10.10 PICSTART Plus Entry Level Development Programmer

The PICSTART Plus development programmer is an easy-to-use, low cost, prototype programmer. It connects to the PC via a COM (RS-232) port. MPLAB Integrated Development Environment software makes using the programmer simple and efficient.

The PICSTART Plus development programmer supports all PICmicro devices with up to 40 pins. Larger pin count devices, such as the PIC16C92X and PIC17C76X, may be supported with an adapter socket. The PICSTART Plus development programmer is CE compliant.

## 10.11 PICDEM 1 Low Cost PICmicro Demonstration Board

The PICDEM 1 demonstration board is a simple board which demonstrates the capabilities of several of Microchip's microcontrollers. The microcontrollers supported are: PIC16C5X (PIC16C54 to PIC16C58A), PIC16C61, PIC16C62X, PIC16C71, PIC16C8X, PIC17C42, PIC17C43 and PIC17C44. All necessary hardware and software is included to run basic demo programs. The user can program the sample microcontrollers provided with the PICDEM 1 demonstration board on a PRO MATE II device programmer, or a PICSTART Plus development programmer, and easily test firmware. The user can also connect the PICDEM 1 demonstration board to the MPLAB ICE in-circuit emulator and download the firmware to the emulator for testing. A prototype area is available for the user to build some additional hardware and connect it to the microcontroller socket(s). Some of the features include an RS-232 interface, a potentiometer for simulated analog input, push button switches and eight LEDs connected to PORTB.

## 10.12 PICDEM 2 Low Cost PIC16CXX Demonstration Board

The PICDEM 2 demonstration board is a simple demonstration board that supports the PIC16C62, PIC16C64, PIC16C65, PIC16C73 and PIC16C74 microcontrollers. All the necessary hardware and software is included to run the basic demonstration programs. The user can program the sample microcontrollers provided with the PICDEM 2 demonstration board on a PRO MATE II device programmer, or a PICSTART Plus development programmer, and easily test firmware. The MPLAB ICE in-circuit emulator may also be used with the PICDEM 2 demonstration board to test firmware. A prototype area has been provided to the user for adding additional hardware and connecting it to the microcontroller socket(s). Some of the features include a RS-232 interface, push button switches, a potentiometer for simulated analog input, a serial EEPROM to demonstrate usage of the $I^2C$™ bus and separate headers for connection to an LCD module and a keypad.

## 10.13 PICDEM 3 Low Cost PIC16CXXX Demonstration Board

The PICDEM 3 demonstration board is a simple demonstration board that supports the PIC16C923 and PIC16C924 in the PLCC package. It will also support future 44-pin PLCC microcontrollers with an LCD Module. All the necessary hardware and software is included to run the basic demonstration programs. The user can program the sample microcontrollers provided with the PICDEM 3 demonstration board on a PRO MATE II device programmer, or a PICSTART Plus development programmer with an adapter socket, and easily test firmware. The MPLAB ICE in-circuit emulator may also be used with the PICDEM 3 demonstration board to test firmware. A prototype area has been provided to the user for adding hardware and connecting it to the microcontroller socket(s). Some of the features include a RS-232 interface, push button switches, a potentiometer for simulated analog input, a thermistor and separate headers for connection to an external LCD module and a keypad. Also provided on the PICDEM 3 demonstration board is a LCD panel, with 4 commons and 12 segments, that is capable of displaying time, temperature and day of the week. The PICDEM 3 demonstration board provides an additional RS-232 interface and Windows software for showing the demultiplexed LCD signals on a PC. A simple serial interface allows the user to construct a hardware demultiplexer for the LCD signals.

## 10.14 PICDEM 17 Demonstration Board

The PICDEM 17 demonstration board is an evaluation board that demonstrates the capabilities of several Microchip microcontrollers, including PIC17C752, PIC17C756A, PIC17C762 and PIC17C766. All necessary hardware is included to run basic demo programs, which are supplied on a 3.5-inch disk. A programmed sample is included and the user may erase it and program it with the other sample programs using the PRO MATE II device programmer, or the PICSTART Plus development programmer, and easily debug and test the sample code. In addition, the PICDEM 17 demonstration board supports downloading of programs to and executing out of external FLASH memory on board. The PICDEM 17 demonstration board is also usable with the MPLAB ICE in-circuit emulator, or the PICMASTER emulator and all of the sample programs can be run and modified using either emulator. Additionally, a generous prototype area is available for user hardware.

## 10.15 KEELOQ Evaluation and Programming Tools

KEELOQ evaluation and programming tools support Microchip's HCS Secure Data Products. The HCS evaluation kit includes a LCD display to show changing codes, a decoder to decode transmissions and a programming interface to program test transmitters.

**TABLE 10-1: DEVELOPMENT TOOLS FROM MICROCHIP**

| | | PIC12CXXX | PIC14000 | PIC16C5X | PIC16C6X | PIC16CXXX | PIC16F62X | PIC16C7X | PIC16C7XX | PIC16C8X | PIC16F8XX | PIC16C9XX | PIC17C4X | PIC17C7XX | PIC18CXX2 | PIC18FXXX | 24CXX/ 25CXX/ 93CXX | HCSXXX | MCRFXXX | MCP2510 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Software Tools | MPLAB® Integrated Development Environment | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | | | | |
| | MPLAB® C17 C Compiler | | | | | | | | | | | | ✓ | ✓ | | | | | | |
| | MPLAB® C18 C Compiler | | | | | | | | | | | | | | ✓ | ✓ | | | | |
| | MPASM™ Assembler/ MPLINK™ Object Linker | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | | |
| Emulators | MPLAB® ICE In-Circuit Emulator | ✓ | ✓ | ✓ | ✓ | ✓ | ✓** | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | | | | |
| | ICEPIC™ In-Circuit Emulator | ✓ | | ✓ | ✓ | ✓ | | ✓ | ✓ | ✓ | | ✓ | | | | | | | | |
| Debugger | MPLAB® ICD In-Circuit Debugger | | | | ✓* | | | ✓* | | | ✓ | | | | | ✓ | | | | |
| Programmers | PICSTART® Plus Entry Level Development Programmer | ✓ | ✓ | ✓ | ✓ | ✓ | ✓** | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | | | | |
| | PRO MATE® II Universal Device Programmer | ✓ | ✓ | ✓ | ✓ | ✓ | ✓** | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | | |
| Demo Boards and Eval Kits | PICDEM™ 1 Demonstration Board | | | ✓ | | ✓ | | ✓† | | ✓ | | | ✓ | | | | | | | |
| | PICDEM™ 2 Demonstration Board | | | | ✓† | | | ✓† | | | | | | | ✓ | ✓ | | | | |
| | PICDEM™ 3 Demonstration Board | | | | | | | | | | | ✓ | | | | | | | | |
| | PICDEM™ 14A Demonstration Board | | ✓ | | | | | | | | | | | | | | | | | |
| | PICDEM™ 17 Demonstration Board | | | | | | | | | | | | | ✓ | | | | | | |
| | KEELOQ® Evaluation Kit | | | | | | | | | | | | | | | | ✓ | | | |
| | KEELOQ® Transponder Kit | | | | | | | | | | | | | | | | ✓ | | | |
| | microID™ Programmer's Kit | | | | | | | | | | | | | | | | | ✓ | | |
| | 125 kHz microID™ Developer's Kit | | | | | | | | | | | | | | | | | ✓ | | |
| | 125 kHz Anticollision microID™ Developer's Kit | | | | | | | | | | | | | | | | | ✓ | | |
| | 13.56 MHz Anticollision microID™ Developer's Kit | | | | | | | | | | | | | | | | | ✓ | | |
| | MCP2510 CAN Developer's Kit | | | | | | | | | | | | | | | | | | | ✓ |

* Contact the Microchip Technology Inc. web site at www.microchip.com for information on how to use the MPLAB® ICD In-Circuit Debugger (DV164001) with PIC16C62, 63, 64, 65, 72, 73, 74, 76, 77.

** Contact Microchip Technology Inc. for availability date.

† Development tool is available on select devices.

**NOTES:**

## 11.0 ELECTRICAL CHARACTERISTICS

### Absolute Maximum Ratings†

Ambient Temperature under bias ..... -40°C to +85°C

Storage Temperature ..... -40°C to +150°C

Absolute Maximum Ratings - PICmicro

Voltage on $V_{DD}$ with respect to $V_{SS}$ ..... 0 to +7.0 V

Absolute Maximum Ratings - Transmitter

Voltage on $V_{DDRF}$ with respect to $V_{SSRF}$ ..... -0.3 to +7.0 V

Voltage on $\overline{MCLR}$ with respect to $V_{SS}$ ..... 0 to +14 V

Voltage on all other PICmicro pins with respect to $V_{SS}$ ..... -0.3 V to ($V_{DD}$ + 0.3 V)

Voltage on all other Transmitter pins with respect to $V_{SSRF}$ ..... -0.3 to ($V_{DDRF}$ +0.3 V)

Max. Current into $RF_{EN}$ pin ..... -1.0 to 1.0 mA

Total Power Dissipation(1) ..... 700 mW

Max. Current out of $V_{SS}$ pin ..... 200 mA

Max. Current into $V_{DD}$ pin ..... 150 mA

Input Clamp Current, $I_{IK}$ ($V_I$ < 0 or $V_I$ > $V_{DD}$) ..... ±20 mA

Output Clamp Current, $I_{OK}$ ($V_O$ < 0 or $V_O$ > $V_{DD}$) ..... ±20 mA

Max. Output Current sunk by any I/O pin ..... 25 mA

Max. Output Current sourced by any I/O pin ..... 25 mA

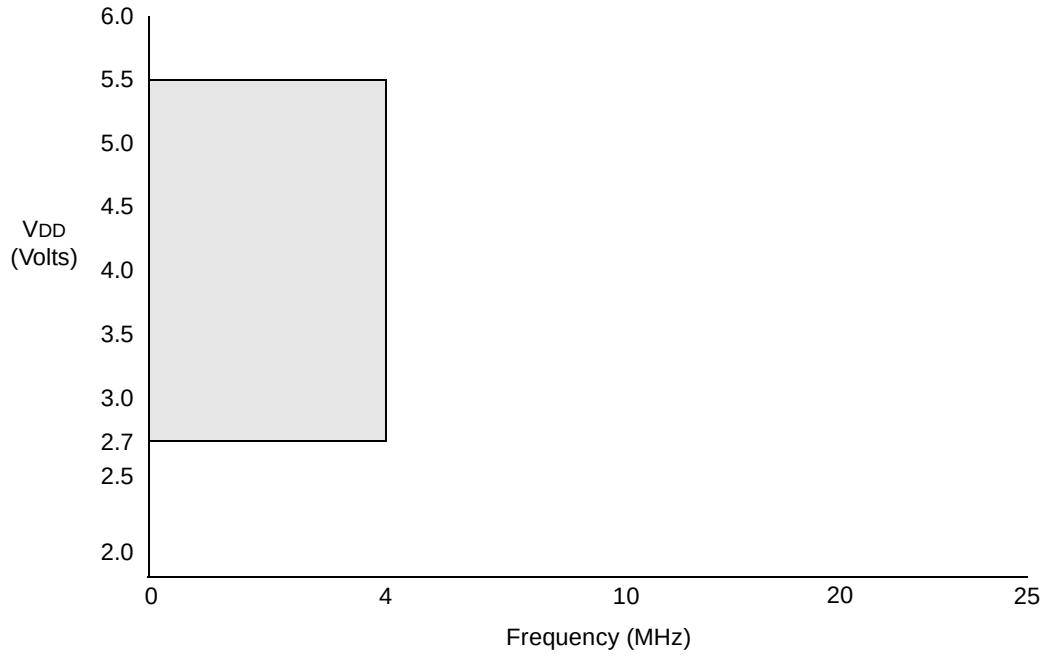
Max. Output Current sourced by I/O port (GPIO) ..... 100 mA

Max. Output Current sunk by I/O port (GPIO ) ..... 100 mA

**Note 1:** Power Dissipation is calculated as follows: $P_{DIS} = V_{DD} \times \{I_{DD} - \sum I_{OH}\} + \sum \{(V_{DD}-V_{OH}) \times I_{OH}\} + \sum(V_{OL} \times I_{OL})$

†NOTICE: Stresses above those listed under "Maximum Ratings" may cause permanent damage to the device. This is a stress rating only and functional operation of the device at those or any other conditions above those indicated in the operation listings of this specification is not implied. Exposure to maximum rating conditions for extended periods may affect device reliability.

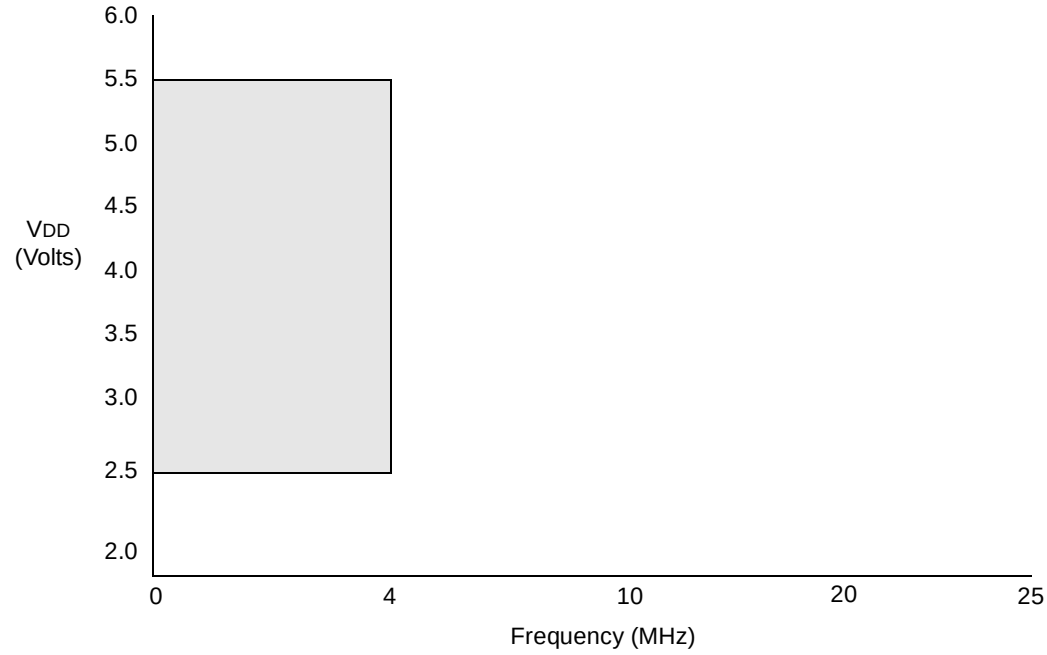
**FIGURE 11-1: rfPIC12C509AG/509AF VOLTAGE-FREQUENCY GRAPH, -40°C ≤ TA ≤ 0°C**




**Note 1:** The shaded region indicates the permissible combinations of voltage and frequency.

**2:** The maximum rated speed of the part limits the permissible combinations of voltage and frequency. Please reference the Product Identification System section for the maximum rated speed of the parts.

**FIGURE 11-2: rfPIC12C509AG/509AF VOLTAGE-FREQUENCY GRAPH, 0°C ≤ TA ≤ +85°C**




**Note 1:** The shaded region indicates the permissible combinations of voltage and frequency.

**2:** The maximum rated speed of the part limits the permissible combinations of voltage and frequency. Please reference the Product Identification System section for the maximum rated speed of the parts.

## 11.1 DC CHARACTERISTICS: rfPIC12C509AG/509AF (Industrial)

| DC Characteristics | | | Standard Operating Conditions (unless otherwise specified)<br>Operating Temperature –40°C ≤ TA ≤ +85°C (industrial) | | | | |
|---|---|---|---|---|---|---|---|
| **Param No.** | **Sym** | **Characteristic** | **Min** | **Typ(1)** | **Max** | **Units** | **Conditions** |
| D001 | VDD | **Supply Voltage** | 2.5 | | 5.5 | V | See Figures 11-1 through 11-2 |
| D002 | VDR | **RAM Data Retention Voltage(2)** | | 1.5* | | V | Device in SLEEP mode |
| D003 | VPOR | **VDD Start Voltage to ensure Power-on Reset** | | VSS | | V | See section on Power-on Reset for details |
| D004 | SVDD | **VDD Rise Rate to ensure Power-on Reset** | 0.05 * | | | V/ms | See section on Power-on Reset for details |
| D010 | IDD | **Supply Current(3)** | - | 0.4 | 0.8 | mA | XT and EXTRC options (Note 4)<br>FOSC = 4 MHz, VDD = 2.5V |
| D010C | | | - | 0.4 | 0.8 | mA | INTRC Option<br>FOSC = 4 MHz, VDD = 2.5V |
| D010A | | | - | 115 | 31 | µA | LP Option, Industrial Temperature<br>FOSC = 32 kHz, VDD = 2.5V, WDT disabled |
| D020<br>D021<br>D021B | IPD | **Power-Down Current (5)** | - | 0.2 | 4 | µA | VDD = 2.5V, Industrial |
| | ΔIWDT | | - | 2.0 | 5 | µA | VDD = 2.5V, Industrial |
| 1A | Fosc | **LP Oscillator Operating Frequency** | 0 | - | 200 | kHz | All temperatures |
| | | **XT Oscillator Operating Frequency** | 0 | - | 4 | MHz | All temperatures |

* These parameters are characterized but not tested.

**Note 1:** Data in the Typical ("Typ") column is based on characterization results at 25°C. This data is for design guidance only and is not tested.

**2:** This is the limit to which VDD can be lowered in SLEEP mode without losing RAM data.

**3:** The supply current is mainly a function of the operating voltage and frequency. Other factors such as bus loading, oscillator type, bus rate, internal code execution pattern, and temperature also have an impact on the current consumption.

a) The test conditions for all IDD measurements in active operation mode are: OSC1 = external square wave, from rail-to-rail; all I/O pins tristated, pulled to Vss, T0CKI = VDD, $\overline{MCLR}$ = VDD; WDT enabled/disabled as specified.

b) For standby current measurements, the conditions are the same, except that the device is in SLEEP mode.

**4:** Does not include current through REXT. The current through the resistor can be estimated by the formula: IR = VDD/2REXT (mA) with REXT in kOhm.

**5:** The power down current in SLEEP mode does not depend on the oscillator type. Power down current is measured with the part in SLEEP mode, with all I/O pins in hi-impedance state and tied to VDD or VSS.

## 11.2 DC CHARACTERISTICS: rfPIC12C509AG/509AF (Industrial)

| DC CHARACTERISTICS | | | **Standard Operating Conditions (unless otherwise specified)**<br>Operating Temperature –40°C ≤ TA ≤ +85°C (industrial)<br>Operating voltage VDD range as described in DC spec Section 11.1. | | | | |
|---|---|---|---|---|---|---|---|
| **Param No.** | **Sym** | **Characteristic** | **Min** | **Typ†** | **Max** | **Units** | **Conditions** |
| | | **Input Low Voltage** | | | | | |
| | VIL | I/O ports | | | | | |
| D030 | | with TTL buffer | Vss | - | 0.8V | V | For 4.5V ≤ VDD ≤ 5.5V |
| | | | Vss | - | 0.15VDD | V | otherwise |
| D031 | | with Schmitt Trigger buffer | Vss | - | 0.2VDD | V | |
| D032 | | $\overline{\text{MCLR}}$, GP2/T0CKI (in EXTRC mode) | Vss | - | 0.2VDD | V | |
| D033 | | OSC1 (in EXTRC mode) | Vss | - | 0.2VDD | V | **Note 1** |
| D033 | | OSC1 (in XT and LP) | Vss | - | 0.3VDD | V | **Note 1** |
| | | **Input High Voltage** | | | | | |
| | VIH | I/O ports | | - | | | |
| D040 | | with TTL buffer | 2.0V | - | VDD | V | 4.5V ≤ VDD ≤ 5.5V |
| D040A | | | 0.25VDD + 0.8V | - | VDD | V | otherwise |
| D041 | | with Schmitt Trigger buffer | 0.8VDD | - | VDD | V | For entire VDD range |
| D042 | | $\overline{\text{MCLR}}$, GP2/T0CKI | 0.8VDD | - | VDD | V | |
| D042A | | OSC1 (XT and LP) | 0.7VDD | - | VDD | V | **Note 1** |
| D043 | | OSC1 (in EXTRC mode) | 0.9VDD | - | VDD | V | |
| D070 | IPUR | GPIO weak pull-up current (Note 4) | 30 | 250 | 400 | µA | VDD = 5V, VPIN = VSS |
| | | **Input Leakage Current** (Notes 2, 3) | | | | | |
| D060 | IIL | I/O ports | - | - | ±1 | µA | Vss ≤ VPIN ≤ VDD, Pin at hi-impedance |
| D061 | | GP3/$\overline{\text{MCLR}}$ (Note 5) | 8 | 130 | 250 | µA | Vss ≤ VPIN ≤ VDD |
| D061A | | GP3/$\overline{\text{MCLR}}$ (Note 6) | - | - | ±5 | µA | Vss ≤ VPIN ≤ VDD |
| D063 | | OSC1 | - | - | ±5 | µA | Vss ≤ VPIN ≤ VDD, XT and LP osc configuration |
| | | **Output Low Voltage** | | | | | |
| D080 | VOL | I/O ports | - | - | 0.6 | V | IOL = 8.5 mA, VDD = 4.5V, –40°C to +85°C |
| D080A | | | - | - | 0.6 | V | IOL = 7.0 mA, VDD = 4.5V, –40°C to +125°C |
| | | **Output High Voltage** | | | | | |
| D090 | VOH | I/O ports (Note 3) | VDD - 0.7 | - | - | V | IOH = -3.0 mA, VDD = 4.5V, –40°C to +85°C |
| D090A | | | VDD - 0.7 | - | - | V | IOH = -2.5 mA, VDD = 4.5V, –40°C to +125°C |
| | | **Capacitive Loading Specs on Output Pins** | | | | | |
| D100 | COSC2 | OSC2 pin | - | - | 15 | pF | In XT and LP modes when external clock is used to drive OSC1. |
| D101 | CIO | All I/O pins | - | - | 50 | pF | |

† Data in "Typ" column is at 5V, 25°C unless otherwise stated. These parameters are for design guidance only and are not tested.

**Note 1:** In EXTRC oscillator configuration, the OSC1/CLKIN pin is a Schmitt Trigger input. It is not recommended that the rfPIC12C509AG be driven with external clock in RC mode.

**2:** The leakage current on the $\overline{\text{MCLR}}$ 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 coming out of the pin.

**4:** Does not include GP3. For GP3 parameters D0061 and D0061A.

**TABLE 11-1: PULL-UP RESISTOR RANGES***

| VDD (Volts) | Temperature (°C) | Min | Typ | Max | Units |
|---|---|---|---|---|---|
| GP0/GP1 | | | | | |
| 2.5 | –40 | 38K | 42K | 63K | Ω |
| | 25 | 42K | 48K | 63K | Ω |
| | 85 | 42K | 49K | 63K | Ω |
| 5.5 | –40 | 15K | 17K | 20K | Ω |
| | 25 | 18K | 20K | 23K | Ω |
| | 85 | 19K | 22K | 25K | Ω |
| GP3(1) | | | | | |
| 2.5 | –40 | 65K | 80K | 850K | Ω |
| | 25 | 80K | 100K | 1150K | Ω |
| | 85 | 85K | 110K | 1300K | Ω |
| 5.5 | –40 | 50K | 60K | 600K | Ω |
| | 25 | 60K | 65K | 750K | Ω |
| | 85 | 65K | 80K | 900K | Ω |

***** These parameters are characterized but not tested.

**Note 1:** The weak pull-up resistor and associated current for the GP3/$\overline{\text{MCLR}}$ pin is non-linear when the respective pin voltage is less than VDD - 1.0V. See parameter D061 for GP3/$\overline{\text{MCLR}}$ pin current specifications.

## 11.3 Reset

When $\overline{\text{MCLR}}$ is asserted, the state of the OSC1/CLKIN and OSC2 pins are as follows:

**TABLE 11-2: CLKIN/CLKOUT PIN STATES WHEN $\overline{\text{MCLR}}$ ASSERTED**

| Oscillator Mode | OSC1/CLKIN Pin | OSC2 Pin |
|---|---|---|
| EXTRC | OSC1 pin is tristated and driven by external circuit | OSC2 pin is driven low |
| INTRC | OSC1 pin is tristate input | OSC2 pin is tristate input |

## 11.4 Timing Parameter Symbology and Load Conditions

The timing parameter symbols have been created following one of the following formats:

1. TppS2ppS
2. TppS

| **T** | | | |
|---|---|---|---|
| F | Frequency | T | Time |

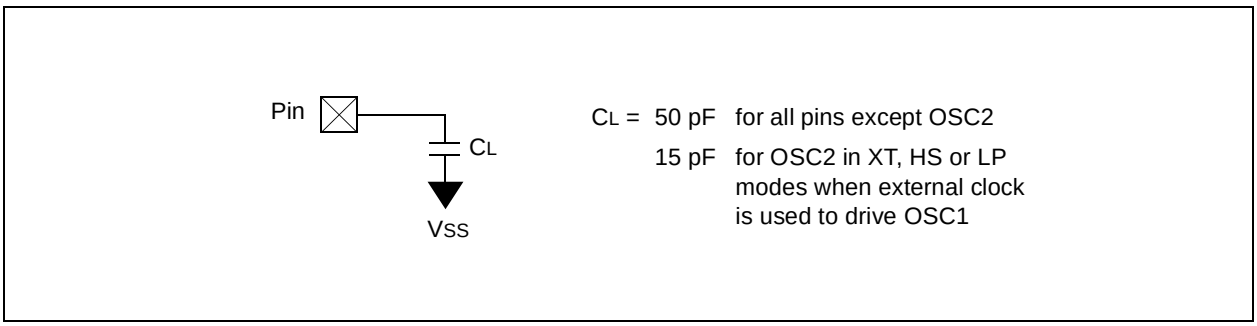
Lowercase subscripts (pp) and their meanings:

| **pp** | | | |
|---|---|---|---|
| 2 | to | mc | $\overline{\text{MCLR}}$ |
| ck | CLKOUT | osc | oscillator |
| cy | cycle time | os | OSC1 |
| drt | device reset timer | t0 | T0CKI |
| io | I/O port | wdt | watchdog timer |

Uppercase letters and their meanings:

| **S** | | | |
|---|---|---|---|
| F | Fall | P | Period |
| H | High | R | Rise |
| I | Invalid (Hi-impedance) | V | Valid |
| L | Low | Z | Hi-impedance |

**FIGURE 11-3: LOAD CONDITIONS**



## 11.5 Timing Diagrams and Specifications

**FIGURE 11-4: EXTERNAL CLOCK TIMING**

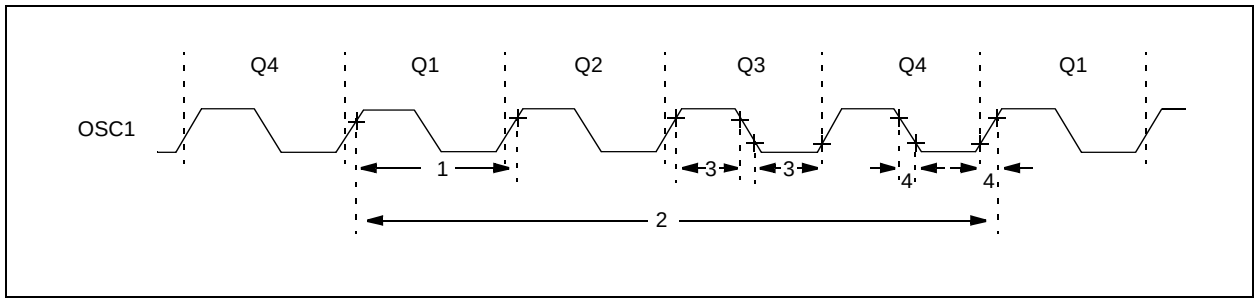



**TABLE 11-3: EXTERNAL CLOCK TIMING REQUIREMENTS**

| AC Characteristics | | Standard Operating Conditions (unless otherwise specified)<br>Operating Temperature –40°C ≤ TA ≤ +85°C (industrial)<br>Operating Voltage VDD range is described in Section 11.1 | | | | | |
|---|---|---|---|---|---|---|---|
| **Parameter No.** | **Sym** | **Characteristic** | **Min** | **Typ(1)** | **Max** | **Units** | **Conditions** |
| | FOSC | **External CLKIN Frequency(2)** | | | | | |
| | | | DC | — | 4 | MHz | XT osc mode |
| | | | DC | — | 200 | kHz | LP osc mode |
| | | **Oscillator Frequency(2)** | DC | — | 4 | MHz | EXTRC osc mode |
| | | | 0.1 | — | 4 | MHz | XT osc mode |
| | | | DC | — | 200 | kHz | LP osc mode |
| 1 | TOSC | **External CLKIN Period(2)** | | | | | |
| | | | 250 | — | — | ns | XT osc mode |
| | | | 5 | — | — | ms | LP osc mode |
| | | **Oscillator Period(2)** | 250 | — | — | ns | EXTRC osc mode |
| | | | 250 | — | 10,000 | ns | XT osc mode |
| | | | 5 | — | — | ms | LP osc mode |
| 2 | Tcy | **Instruction Cycle Time(3)** | — | 4/FOSC | — | — | |
| 3 | TosL, TosH | **Clock in (OSC1) Low or High Time** | 50* | — | — | ns | XT oscillator |
| | | | 2* | — | — | ms | LP oscillator |
| 4 | TosR, TosF | **Clock in (OSC1) Rise or Fall Time** | — | — | 25* | ns | XT oscillator |
| | | | — | — | 50* | ns | LP oscillator |

* These parameters are characterized but not tested.

**Note 1:** Data in the Typical ("Typ") column is at 5V, 25°C unless otherwise stated. These parameters are for design guidance only and are not tested.

**2:** 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.
When an external clock input is used, the "max" cycle time limit is "DC" (no clock) for all devices.

**3:** Instruction cycle period (TCY) equals four times the input oscillator time base period.

**TABLE 11-4: CALIBRATED INTERNAL RC FREQUENCIES**

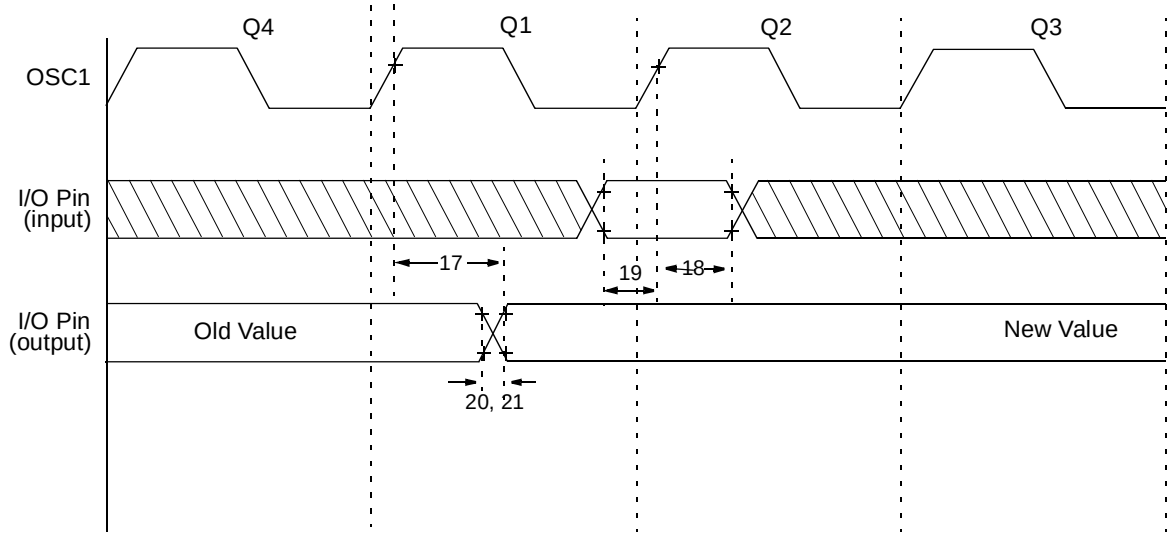
| AC Characteristics | | **Standard Operating Conditions (unless otherwise specified)**<br>Operating Temperature –40°C ≤ TA ≤ +85°C (industrial)<br>Operating Voltage VDD range is described in Section 10.1 | | | | | |
|---|---|---|---|---|---|---|---|
| **Parameter No.** | **Sym** | **Characteristic** | **Min*** | **Typ(1)** | **Max*** | **Units** | **Conditions** |
| | | **Internal Calibrated RC Frequency** | 3.65 | 4.00 | 4.28 | MHz | VDD = 5.0V |
| | | **Internal Calibrated RC Frequency** | 3.55 | — | 4.31 | MHz | VDD = 2.5V |

* These parameters are characterized but not tested.

**Note 1:** Data in the Typical ("Typ") column is at 5V, 25°C unless otherwise stated. These parameters are for design guidance only and are not tested.

**FIGURE 11-5: I/O TIMING**




**Note:** All tests must be done with specified capacitive loads (see data sheet) 50 pF on I/O pins and CLKOUT.

**TABLE 11-5: TIMING REQUIREMENTS**

| **AC Characteristics** | | **Standard Operating Conditions (unless otherwise specified)**<br>Operating Temperature –40°C ≤ TA ≤ +85°C (industrial)<br>Operating Voltage VDD range is described in Section 11.1 | | | | |
|---|---|---|---|---|---|---|
| **Parameter No.** | **Sym** | **Characteristic** | **Min** | **Typ(1)** | **Max** | **Units** |
| 17 | TosH2ioV | OSC1↑ (Q1 cycle) to Port out valid(3) | — | — | 100* | ns |
| 18 | TosH2ioI | OSC1↑ (Q2 cycle) to Port input invalid (I/O in hold time) | TBD | — | — | ns |
| 19 | TioV2osH | Port input valid to OSC1↑ (I/O in setup time) | TBD | — | — | ns |
| 20 | TioR | Port output rise time(2, 3) | — | 10 | 25** | ns |
| 21 | TioF | Port output fall time(2, 3) | — | 10 | 25** | ns |

* These parameters are characterized but not tested.

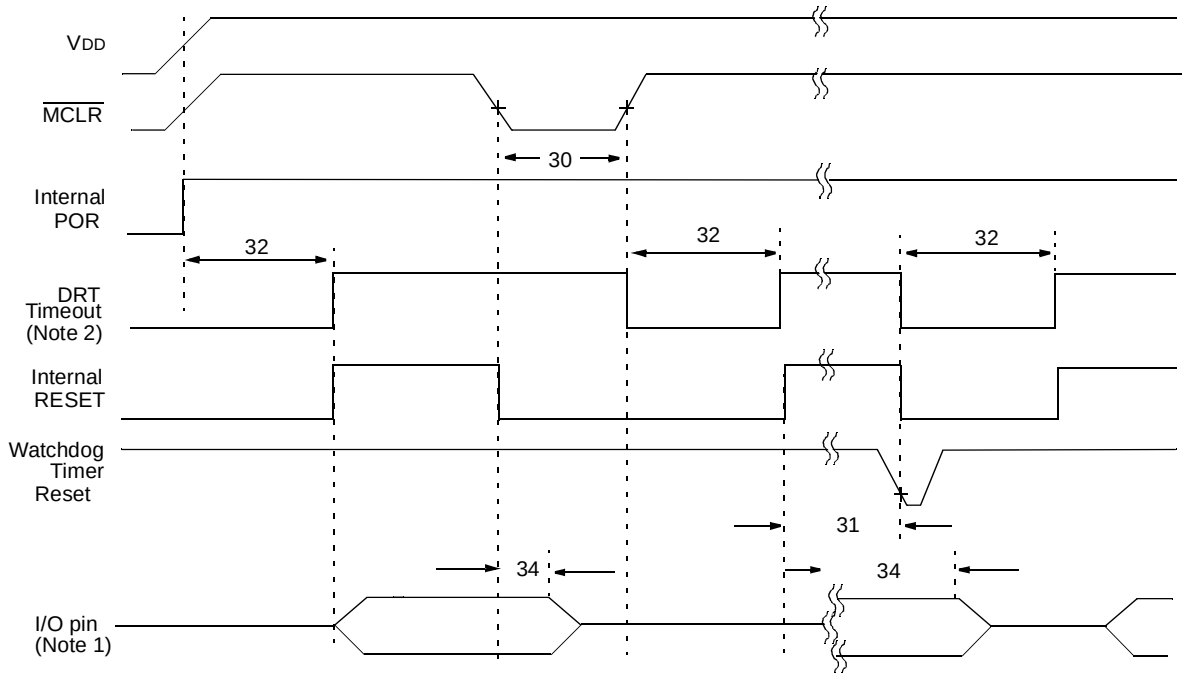
** These parameters are design targets and are not tested. No characterization data available at this time.

**Note 1:** Data in the Typical ("Typ") column is at 5V, 25°C unless otherwise stated. These parameters are for design guidance only and are not tested.

**2:** Measurements are taken in EXTRC mode.

**3:** See Figure 11-3 for loading conditions.

**FIGURE 11-6: RESET, WATCHDOG TIMER AND DEVICE RESET TIMER**




**Note 1:** I/O pins must be taken out of hi-impedance mode by enabling the output drivers in software.
**2:** Runs in MCLR or WDT Reset only in XT and LP modes.

**TABLE 11-6: RESET, WATCHDOG TIMER, AND DEVICE RESET TIMER**

| **AC Characteristics** | | **Standard Operating Conditions (unless otherwise specified)**<br>Operating Temperature –40°C ≤ TA ≤ +85°C (industrial)<br>Operating Voltage VDD range is described in Section 11.1 | | | | | |
|---|---|---|---|---|---|---|---|
| **Parameter No.** | **Sym** | **Characteristic** | **Min** | **Typ(1)** | **Max** | **Units** | **Conditions** |
| 30 | TmcL | $\overline{\text{MCLR}}$ Pulse Width (low) | 2000* | — | — | ns | VDD = 5 V |
| 31 | Twdt | Watchdog Timer Time-out Period (No Prescaler) | 9* | 18* | 30* | ms | VDD = 5 V (Commercial) |
| 32 | TDRT | Device Reset Timer Period(2) | 9* | 18* | 30* | ms | VDD = 5 V (Commercial) |
| 34 | Tioz | I/O Hi-impedance from $\overline{\text{MCLR}}$ Low | — | — | 2000* | ns | |

* These parameters are characterized but not tested.

**Note 1:** Data in the Typical ("Typ") column is at 5V, 25°C unless otherwise stated. These parameters are for design guidance only and are not tested.
**2:** See Table 13-6.

**TABLE 11-7: DRT (DEVICE RESET TIMER PERIOD)**

| Oscillator Configuration | POR Reset | Subsequent RESETS |
|---|---|---|
| IntRC & ExtRC | 18 ms (typical)(1) | 300 µs (typical)(1) |
| XT & LP | 18 ms (typical)(1) | 18 ms (typical)(1) |

**Note 1:** Data in the Typical ("Typ") column is at 5V, 25°C unless otherwise stated. These parameters are for design guidance only and are not tested.

**FIGURE 11-7: TIMER0 CLOCK TIMINGS**

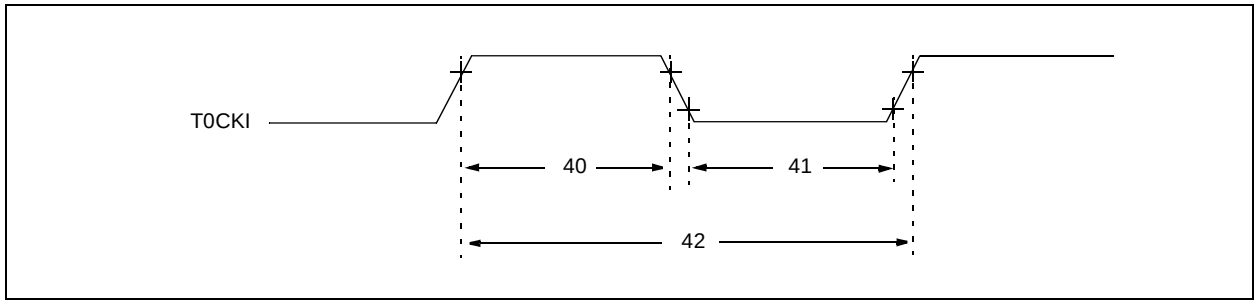



**TABLE 11-8: TIMER0 CLOCK REQUIREMENTS**

| AC Characteristics | | | Standard Operating Conditions (unless otherwise specified)<br>Operating Temperature –40°C ≤ TA ≤ +85°C (industrial)<br>Operating Voltage VDD range is described in Section 11.1. | | | | |
|---|---|---|---|---|---|---|---|
| **Parameter No.** | **Sym** | **Characteristic** | **Min** | **Typ(1)** | **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 | 20 or $\frac{T_{CY} + 40}{N}$* | — | — | ns | Whichever is greater. N =Prescale Value (1, 2, 4,..., 256) |

* These parameters are characterized but not tested.

**Note 1:** Data in the Typical ("Typ") column is at 5V, 25°C unless otherwise stated. These parameters are for design guidance only and are not tested.

## 11.6 Transmitter Characteristics: rfPIC12C509AG/509AF (Industrial)

### TABLE 11-9: TRANSMITTER DC CHARACTERISTICS*

| DC CHARACTERISTICS | | | Standard Operating Conditions (unless otherwise stated)<br>Operating Temperature -40°C ≤ TA ≤ +85°C | | | | |
|---|---|---|---|---|---|---|---|
| **Param No.** | **Sym** | **Characteristic** | **Min** | **Typ†** | **Max** | **Units** | **Conditions** |
| | VDDRF | Supply Voltage | 2.1 | — | 5.5 | V | |
| | IPDRF | Power-Down Current | — | 0.05 | 0.1 | µA | RFEN = 0 |
| | IDDRF | Supply Current | 4.8 | — | 11.5 | mA | **Note 1** |
| | VILRF | Input Low Voltage | -0.3 | — | 0.3VSSRF | V | **Note 2** |
| | VIHRF | Input High Voltage | 0.7VSSRF | — | VSSRF + 0.3 | V | **Note 2** |
| | IILRF | Input Leakage Current | -1 | — | 1 | µA | |

* These parameters are characterized but not tested.

† Data in "Typ" column is at 3V, 25°C unless otherwise stated. These parameters are for design guidance only and are not tested.

**Note 1:** Depends on output power selection. See Table 7-5.

**Note 2:** Applies to RFEN pin.

### TABLE 11-10: TRANSMITTER AC CHARACTERISTICS*

| AC CHARACTERISTICS | | | Standard Operating Conditions (unless otherwise stated)<br>Operating Temperature -40°C ≤ TA ≤ +85°C | | | | |
|---|---|---|---|---|---|---|---|
| **Param No.** | **Sym** | **Characteristic** | **Min** | **Typ†** | **Max** | **Units** | **Conditions** |
| | $f_{xtal}$ | Crystal Frequency | 9.69 | — | 15 | MHz | |
| | $f_{transmit}$ | Transmit Frequency | 310 | — | 480 | MHz | Fixed, set by $f_{xtal}$ |
| | fCLKOUT | CLKOUT Frequency | 2.42 | — | 3.75 | MHz | Fixed, set by $f_{xtal}$ |
| | $P_o$ | Transmit Output Power | -12 | — | +2 | dBm | See Table 7-5 |
| | fASK | ASK Data Rate | — | — | 40 | kHz | |
| | fFSK | FSK Data Rate | — | — | 20 | kHz | |
| | PREF | Reference Spurs (1) | — | -44 | — | dBm | $f_{transmit} \pm f_{xtal}$ |
| | PCLK | Clock Spurs (1) | — | -44 | — | dBm | $f_{transmit} \pm f_{CLKOUT}$ |
| | PHARM | Harmonic Content | — | -40 | — | dBm | $2f_{transmit}$, $3f_{transmit}$, $4f_{transmit,...}$ |
| | POFF | Spurious Output Signal | — | -60 | — | dBm | Vps ≤ 0.1V |
| | PN | Phase Noise | — | -87 | — | dBc/Hz | $f_{transmit}$ ± 500 kHz |
| | KVCO | VCO Gain | — | 100 | — | MHz/V | |
| | ICP | Charge Pump Current | — | ±260 | — | µA | |
| | VCLKOUT | Clock Voltage Swing | — | 2 | — | VPP | $C_{load}$ = 5 pF |
| | $t_{on}$ | Start-up Time | — | 0.9 | — | ms | **Note 2** |

* These parameters are characterized but not tested.

† Data in "Typ" column is at 3V, 25°C unless otherwise stated. These parameters are for design guidance only and are not tested.

**Note 1:** Values dependent on PLL loop filter values.

**Note 2:** $t_{on}$ equals crystal oscillator and PLL start up time.

## 12.0 DC AND AC CHARACTERISTICS

The graphs and tables provided following this note are a statistical summary based on a limited number of samples and are provided for informational purposes only. The performance characteristics listed herein are not tested or guaranteed. In some graphs or tables, the data presented may be outside the specified operating range (e.g., outside specified power supply range) and therefore outside the warranted range.

"Typical" represents the mean of the distribution at 25°C. "Maximum" or "minimum" represents (mean + 3σ) or (mean - 3σ) respectively, where σ is a standard deviation over the whole temperature range.

**FIGURE 12-1: CALIBRATED INTERNAL RC FREQUENCY RANGE VS. TEMPERATURE ($V_{DD}$ = 5.0V) (INTERNAL RC IS CALIBRATED TO 25°C, 5.0V)**

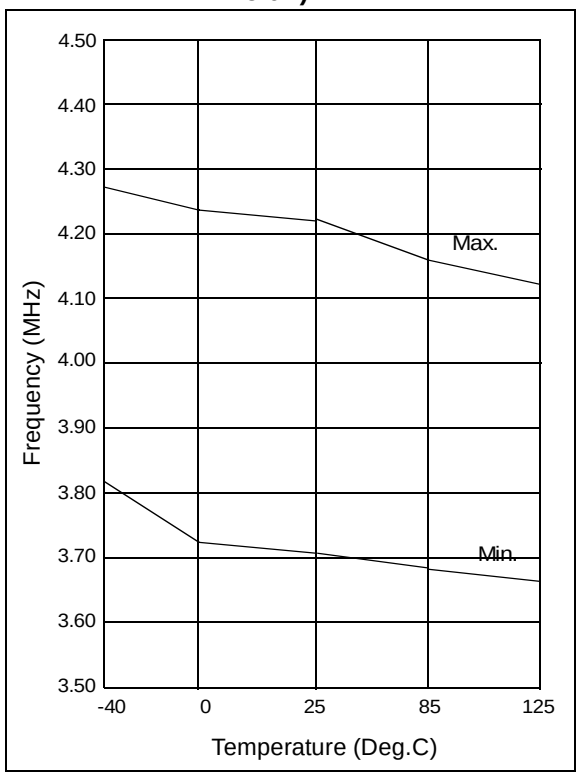



**FIGURE 12-2: CALIBRATED INTERNAL RC FREQUENCY RANGE VS. TEMPERATURE ($V_{DD}$ = 2.5V) (INTERNAL RC IS CALIBRATED TO 25°C, 5.0V)**

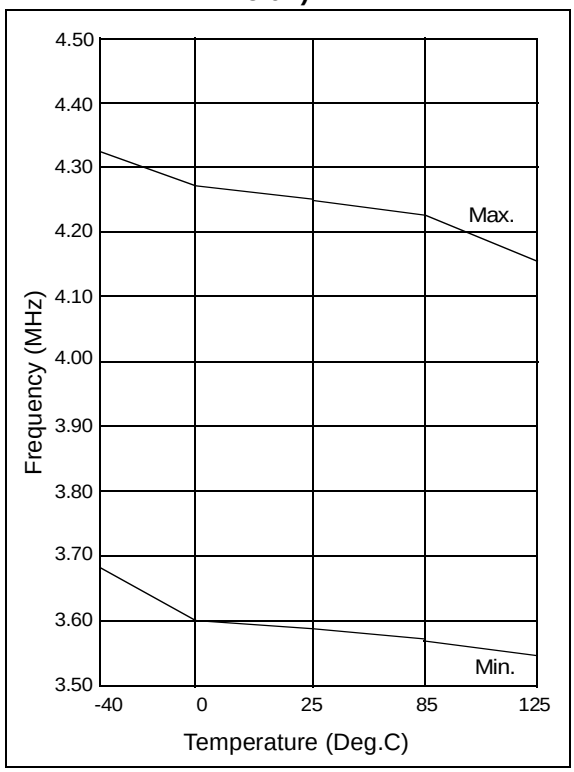


**TABLE 12-1: DYNAMIC IDD (TYPICAL) - WDT ENABLED, 25°C**

| Oscillator | Frequency | VDD =3.0V | VDD = 5.5V |
|---|---|---|---|
| External RC | 4 MHz | 240 µA* | 800 µA* |
| Internal RC | 4 MHz | 320 µA | 800 µA |
| XT | 4 MHz | 300 µA | 800 µA |
| LP | 32 KHz | 19 µA | 50 µA |

*Does not include current through external R&C.

**FIGURE 12-3: TYPICAL IDD VS. VDD (WDT DIS, 25°C, FREQUENCY = 4MHz)**

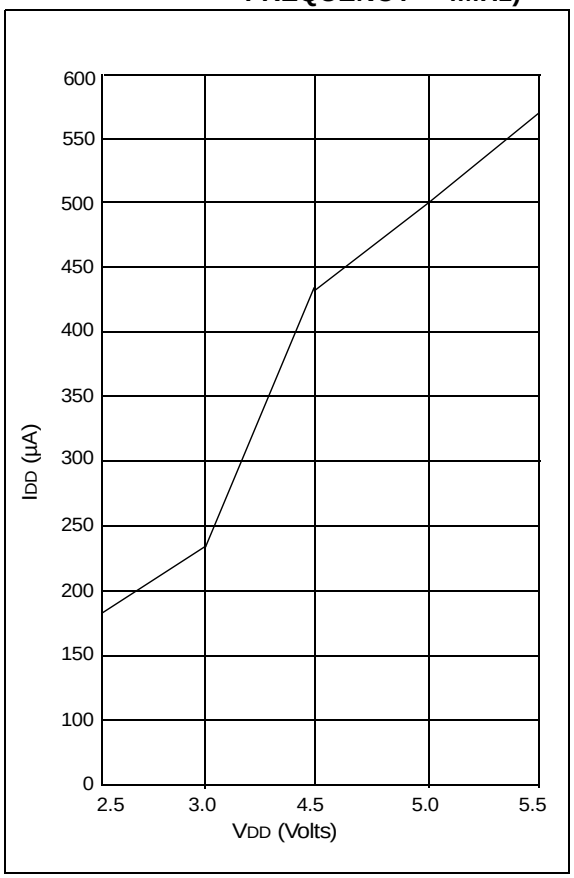



**FIGURE 12-4: TYPICAL IDD VS. FREQUENCY (WDT DIS, 25°C, VDD = 5.5V)**

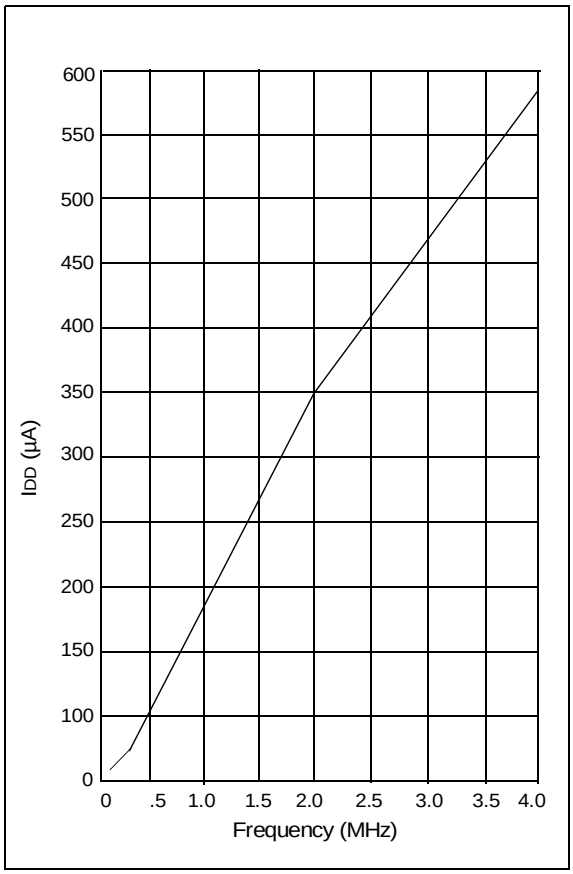


**FIGURE 12-5: WDT TIMER TIME-OUT PERIOD vs. VDD**

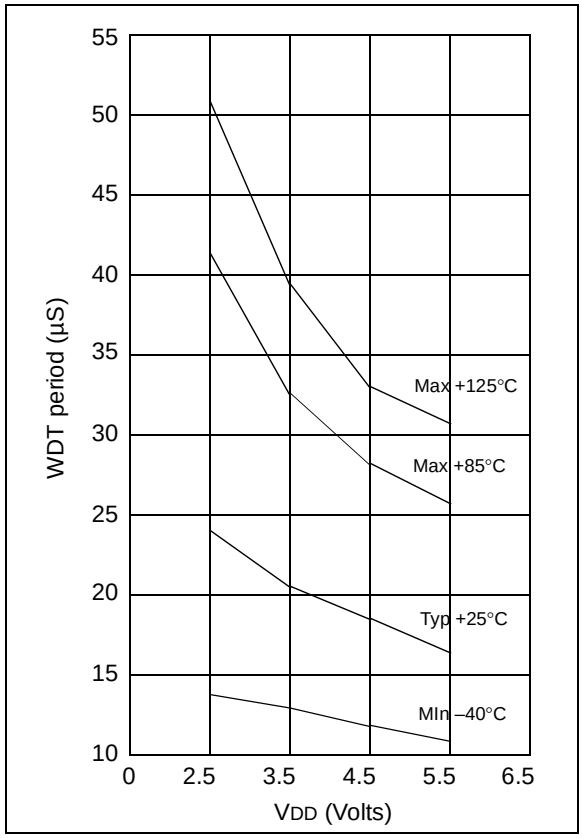



**FIGURE 12-6: SHORT DRT PERIOD VS. VDD**

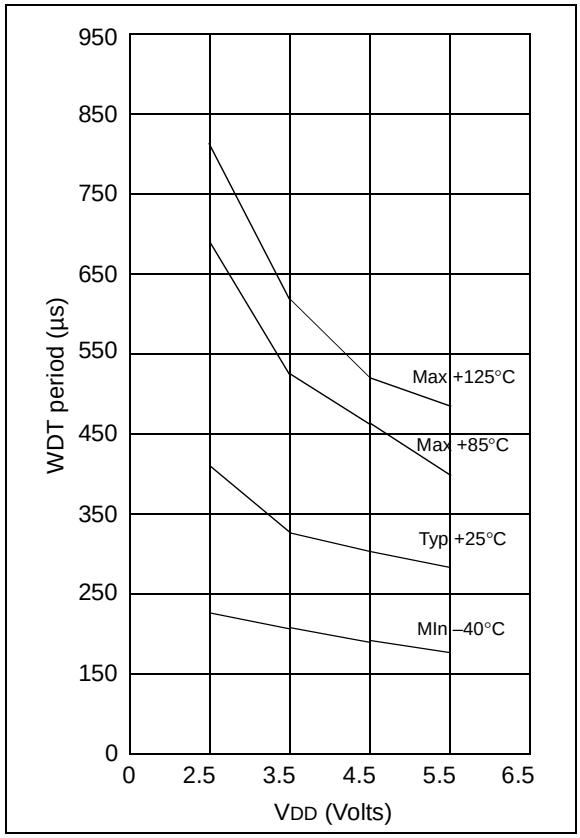



**FIGURE 12-7: IOH vs. VOH, VDD = 2.5 V**

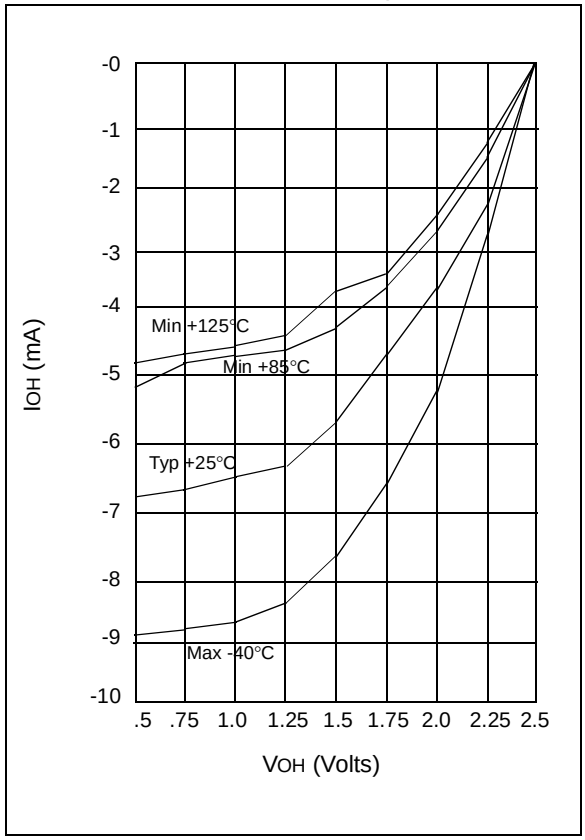



**FIGURE 12-8: IOH vs. VOH, VDD = 3.5 V**

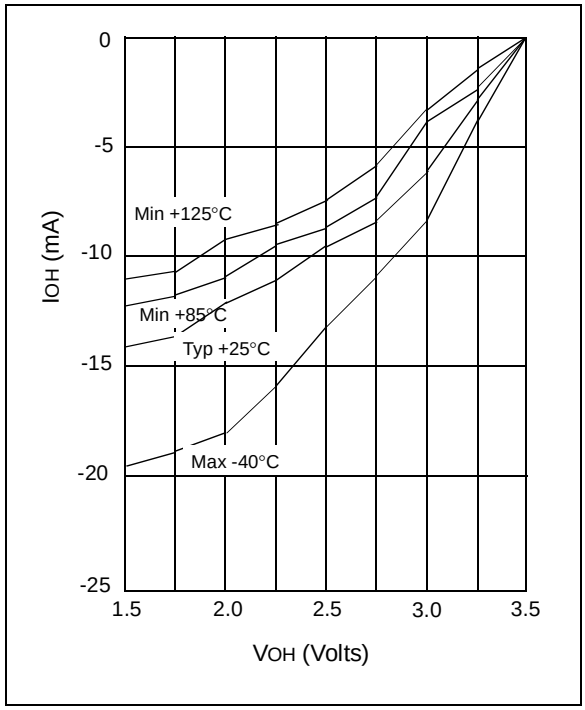


**FIGURE 12-9: IOL vs. VOL, VDD = 2.5 V**

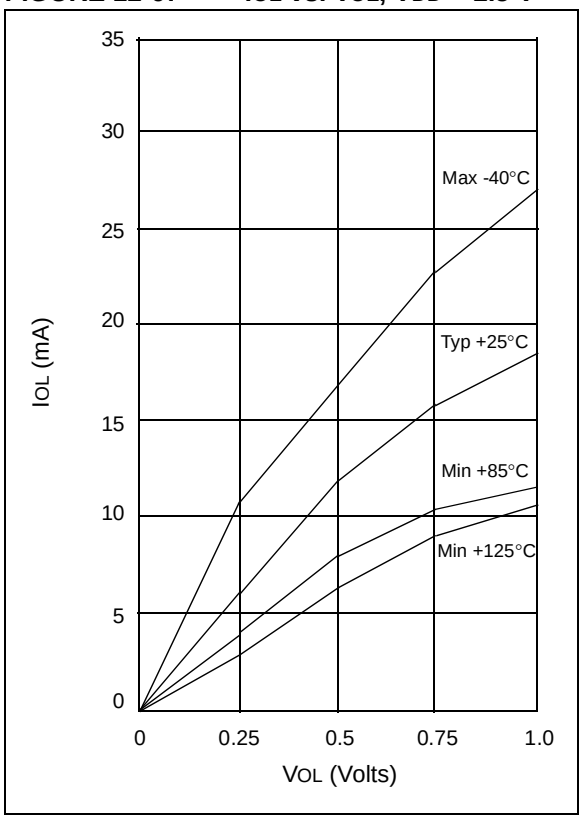



**FIGURE 12-10: IOL vs. VOL, VDD = 3.5 V**

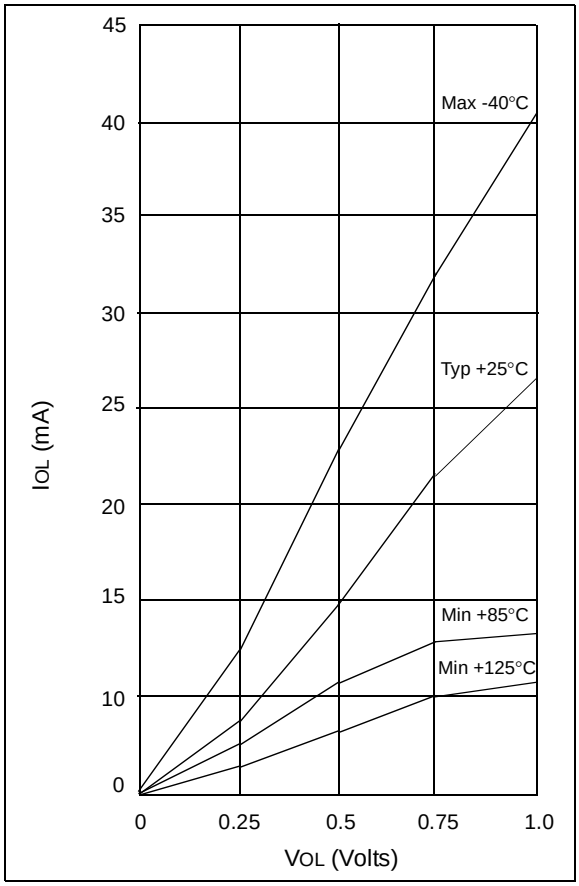



**FIGURE 12-11: IOH vs. VOH, VDD = 5.5 V**

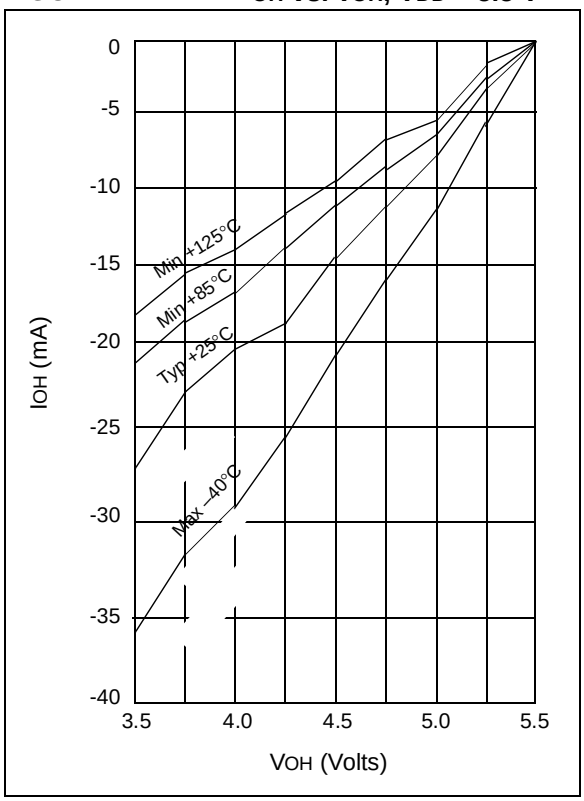



**FIGURE 12-12: IOL vs. VOL, VDD = 5.5 V**

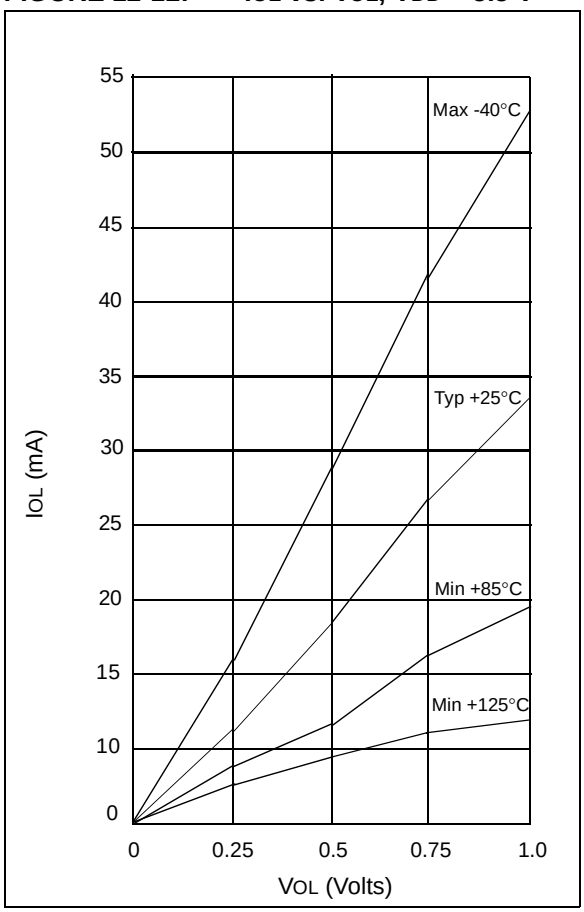


**FIGURE 12-13: TYPICAL IPD VS. VDD, WATCHDOG DISABLED (25°C)**

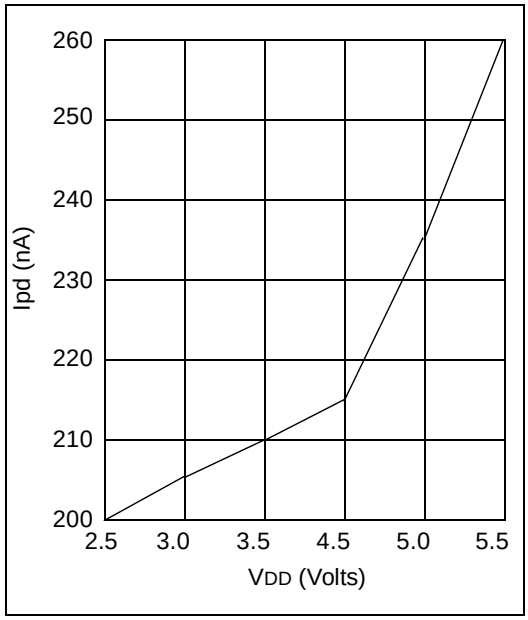



**FIGURE 12-14: VTH (INPUT THRESHOLD VOLTAGE) OF GPIO PINS VS. VDD**

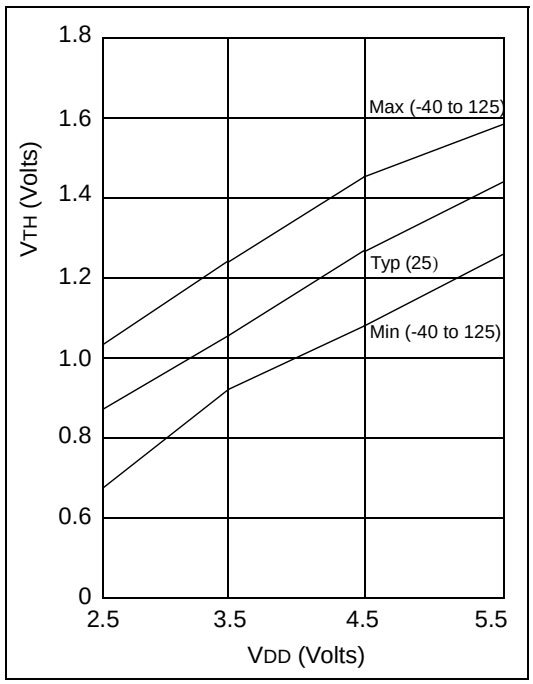
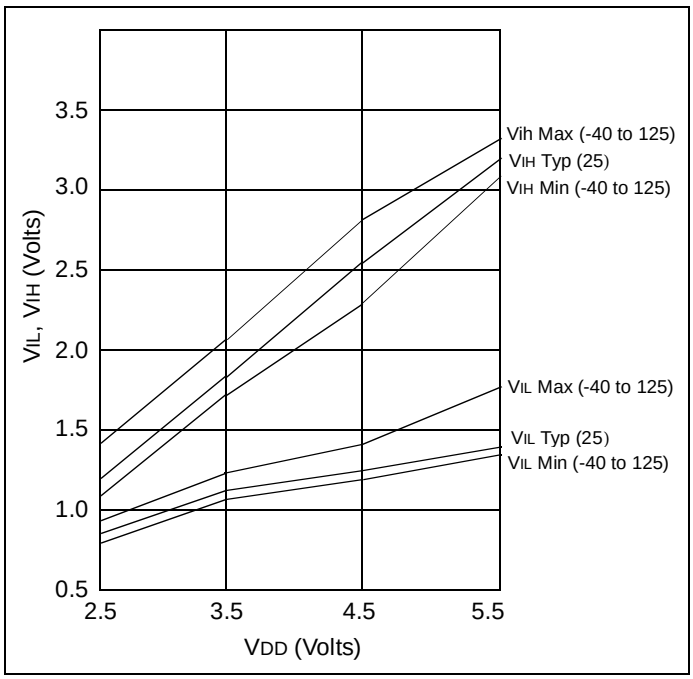


**FIGURE 12-15: VIL, VIH OF NMCLR, AND T0CKI VS. VDD**

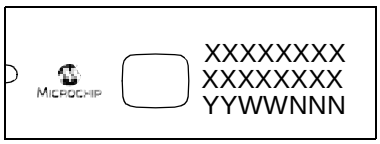


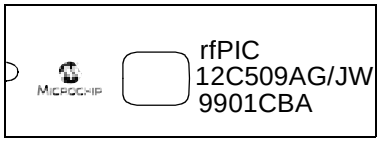
## 13.0 PACKAGING INFORMATION

### 13.1 Package Marking Information

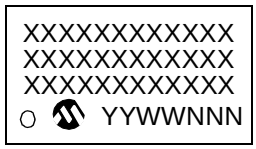
**18-Lead CERDIP Windowed**



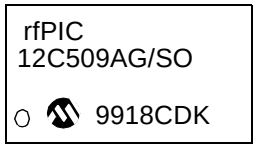

**Example**



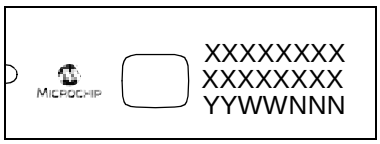

**18-Lead SOIC (.300")**



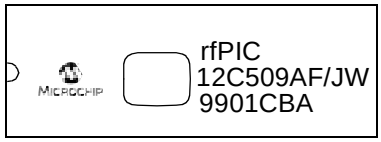

**Example**



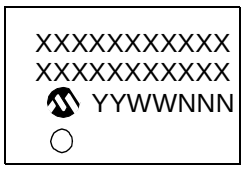

**20-Lead CERDIP Windowed**



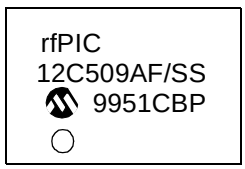

**Example**




**20-Lead SSOP**




**Example**



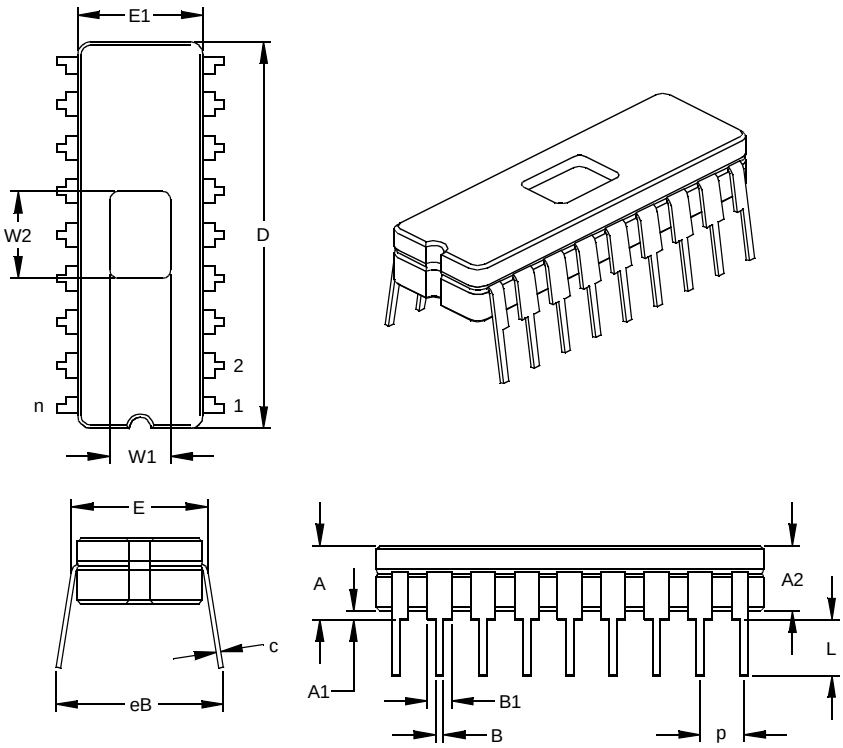

| | | |
|---|---|---|
| **Legend:** | XX...X | Customer specific information* |
| | Y | Year code (last digit of calendar year) |
| | YY | Year code (last 2 digits of calendar year) |
| | WW | Week code (week of January 1 is week '01') |
| | NNN | Alphanumeric traceability code |

**Note**: In the event the full Microchip part number cannot be marked on one line, it will be carried over to the next line thus limiting the number of available characters for customer specific information.

***** Standard PICmicro device marking consists of Microchip part number, year code, week code, and traceability code. For PICmicro device marking beyond this, certain price adders apply. Please check with your Microchip Sales Office. For QTP devices, any special marking adders are included in QTP price.

**Package Type: 18-Lead CERDIP**

**18-Lead Ceramic Dual In-line with Window (JW) - 300 mil (CERDIP)**



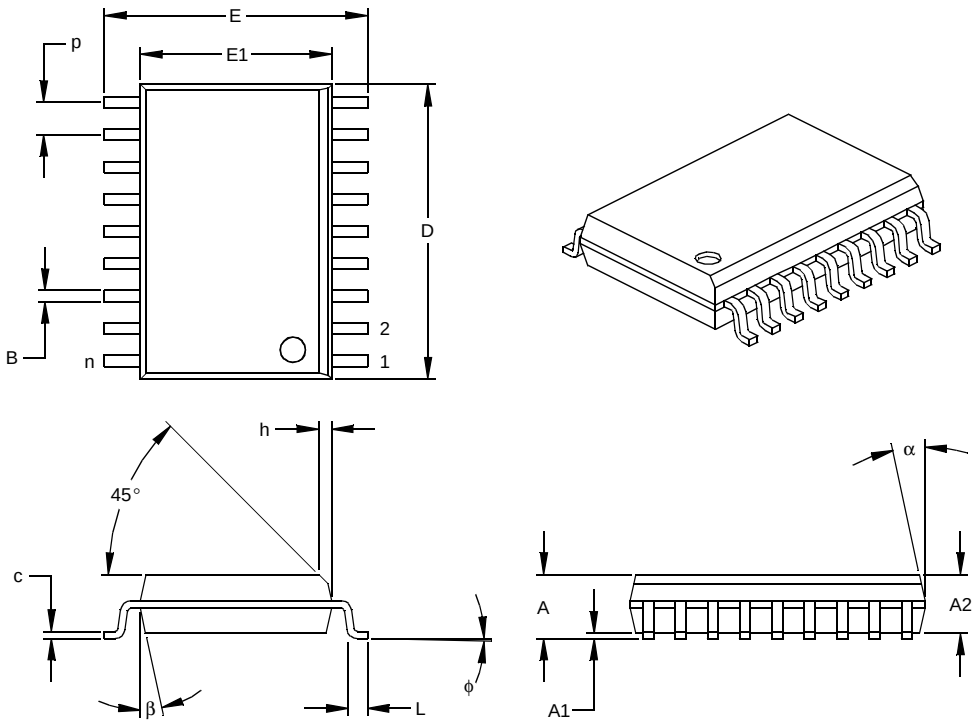

| Units | | INCHES* | | | MILLIMETERS | | |
|---|---|---|---|---|---|---|---|
| Dimension Limits | | MIN | NOM | MAX | MIN | NOM | MAX |
| Number of Pins | n | | 18 | | | 18 | |
| Pitch | p | | .100 | | | 2.54 | |
| Top to Seating Plane | A | .170 | .183 | .195 | 4.32 | 4.64 | 4.95 |
| Ceramic Package Height | A2 | .155 | .160 | .165 | 3.94 | 4.06 | 4.19 |
| Standoff | A1 | .015 | .023 | .030 | 0.38 | 0.57 | 0.76 |
| Shoulder to Shoulder Width | E | .300 | .313 | .325 | 7.62 | 7.94 | 8.26 |
| Ceramic Pkg. Width | E1 | .285 | .290 | .295 | 7.24 | 7.37 | 7.49 |
| Overall Length | D | .880 | .900 | .920 | 22.35 | 22.86 | 23.37 |
| Tip to Seating Plane | L | .125 | .138 | .150 | 3.18 | 3.49 | 3.81 |
| Lead Thickness | c | .008 | .010 | .012 | 0.20 | 0.25 | 0.30 |
| Upper Lead Width | B1 | .050 | .055 | .060 | 1.27 | 1.40 | 1.52 |
| Lower Lead Width | B | .016 | .019 | .021 | 0.41 | 0.47 | 0.53 |
| Overall Row Spacing § | eB | .345 | .385 | .425 | 8.76 | 9.78 | 10.80 |
| Window Width | W1 | .130 | .140 | .150 | 3.30 | 3.56 | 3.81 |
| Window Length | W2 | .190 | .200 | .210 | 4.83 | 5.08 | 5.33 |

* Controlling Parameter
§ Significant Characteristic
JEDEC Equivalent: MO-036
Drawing No. C04-010

**Package Type: 18-Lead SOIC - 300 mil**

**18-Lead Plastic Small Outline (SO) - Wide, 300 mil (SOIC)**



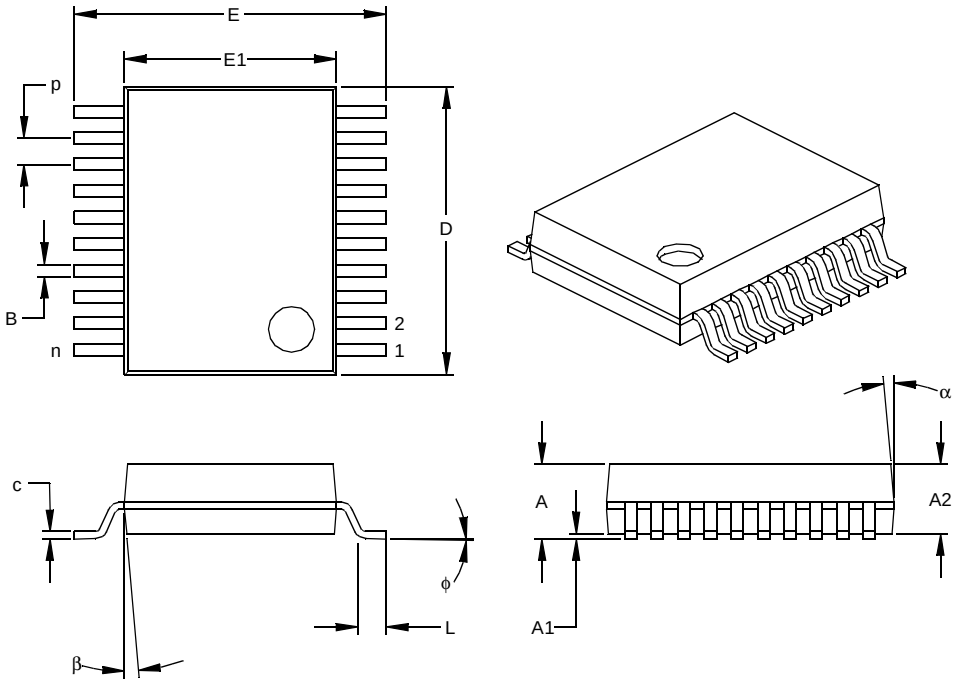

| Units | | INCHES* | | | MILLIMETERS | | |
|---|---|---|---|---|---|---|---|
| Dimension Limits | | MIN | NOM | MAX | MIN | NOM | MAX |
| Number of Pins | n | | 18 | | | 18 | |
| Pitch | p | | .050 | | | 1.27 | |
| Overall Height | A | .093 | .099 | .104 | 2.36 | 2.50 | 2.64 |
| Molded Package Thickness | A2 | .088 | .091 | .094 | 2.24 | 2.31 | 2.39 |
| Standoff § | A1 | .004 | .008 | .012 | 0.10 | 0.20 | 0.30 |
| Overall Width | E | .394 | .407 | .420 | 10.01 | 10.34 | 10.67 |
| Molded Package Width | E1 | .291 | .295 | .299 | 7.39 | 7.49 | 7.59 |
| Overall Length | D | .446 | .454 | .462 | 11.33 | 11.53 | 11.73 |
| Chamfer Distance | h | .010 | .020 | .029 | 0.25 | 0.50 | 0.74 |
| Foot Length | L | .016 | .033 | .050 | 0.41 | 0.84 | 1.27 |
| Foot Angle | ϕ | 0 | 4 | 8 | 0 | 4 | 8 |
| Lead Thickness | c | .009 | .011 | .012 | 0.23 | 0.27 | 0.30 |
| Lead Width | B | .014 | .017 | .020 | 0.36 | 0.42 | 0.51 |
| Mold Draft Angle Top | α | 0 | 12 | 15 | 0 | 12 | 15 |
| Mold Draft Angle Bottom | β | 0 | 12 | 15 | 0 | 12 | 15 |

* Controlling Parameter
§ Significant Characteristic

Notes:
Dimensions D and E1 do not include mold flash or protrusions. Mold flash or protrusions shall not exceed .010” (0.254mm) per side.
JEDEC Equivalent: MS-013
Drawing No. C04-051

**Package Type: 20-Lead SSOP**

**20-Lead Plastic Shrink Small Outline (SS) - 209 mil, 5.30 mm (SSOP)**




| Units | | INCHES* | | | MILLIMETERS | | |
|---|---|---|---|---|---|---|---|
| Dimension Limits | | MIN | NOM | MAX | MIN | NOM | MAX |
| Number of Pins | n | | 20 | | | 20 | |
| Pitch | p | | .026 | | | 0.65 | |
| Overall Height | A | .068 | .073 | .078 | 1.73 | 1.85 | 1.98 |
| Molded Package Thickness | A2 | .064 | .068 | .072 | 1.63 | 1.73 | 1.83 |
| Standoff § | A1 | .002 | .006 | .010 | 0.05 | 0.15 | 0.25 |
| Overall Width | E | .299 | .309 | .322 | 7.59 | 7.85 | 8.18 |
| Molded Package Width | E1 | .201 | .207 | .212 | 5.11 | 5.25 | 5.38 |
| Overall Length | D | .278 | .284 | .289 | 7.06 | 7.20 | 7.34 |
| Foot Length | L | .022 | .030 | .037 | 0.56 | 0.75 | 0.94 |
| Lead Thickness | c | .004 | .007 | .010 | 0.10 | 0.18 | 0.25 |
| Foot Angle | φ | 0 | 4 | 8 | 0.00 | 101.60 | 203.20 |
| Lead Width | B | .010 | .013 | .015 | 0.25 | 0.32 | 0.38 |
| Mold Draft Angle Top | α | 0 | 5 | 10 | 0 | 5 | 10 |
| Mold Draft Angle Bottom | β | 0 | 5 | 10 | 0 | 5 | 10 |

* Controlling Parameter
§ Significant Characteristic

Notes:
Dimensions D and E1 do not include mold flash or protrusions. Mold flash or protrusions shall not exceed .010” (0.254mm) per side.
JEDEC Equivalent: MO-150
Drawing No. C04-072

## INDEX

## ON-LINE SUPPORT

Microchip provides on-line support on the Microchip World Wide Web (WWW) site.

The web site is used by Microchip as a means to make files and information easily available to customers. To view the site, the user must have access to the Internet and a web browser, such as Netscape or Microsoft Explorer. Files are also available for FTP download from our FTP site.

### Connecting to the Microchip Internet Web Site

The Microchip web site is available by using your favorite Internet browser to attach to:

**www.microchip.com**

The file transfer site is available by using an FTP service to connect to:

**ftp://ftp.microchip.com**

The web site and file transfer site provide a variety of services. Users may download files for the latest Development Tools, Data Sheets, Application Notes, User's Guides, Articles and Sample Programs. A variety of Microchip specific business information is also available, including listings of Microchip sales offices, distributors and factory representatives. Other data available for consideration is:

- Latest Microchip Press Releases
- Technical Support Section with Frequently Asked Questions
- Design Tips
- Device Errata
- Job Postings
- Microchip Consultant Program Member Listing
- Links to other useful web sites related to Microchip Products
- Conferences for products, Development Systems, technical information and more
- Listing of seminars and events

### Systems Information and Upgrade Hot Line

The Systems Information and Upgrade Line provides system users a listing of the latest versions of all of Microchip's development systems software products. Plus, this line provides information on how customers can receive any currently available upgrade kits.The Hot Line Numbers are:

1-800-755-2345 for U.S. and most of Canada, and

1-480-792-7302 for the rest of the world.

013001

## READER RESPONSE

It is our intention to provide you with the best documentation possible to ensure successful use of your Microchip product. If you wish to provide your comments on organization, clarity, subject matter, and ways in which our documentation can better serve you, please FAX your comments to the Technical Publications Manager at (480) 792-4150.

Please list the following information, and use this outline to provide us with your comments about this Data Sheet.

To: Technical Publications Manager Total Pages Sent

RE: Reader Response

From: Name _____

Company _____

Address _____

City / State / ZIP / Country _____

Telephone: (_____) _____ - _____ FAX: (_____) _____ - _____

Application (optional):

Would you like a reply? ___Y ___N

Device: **rfPIC12C509AG/509AF** Literature Number: **DS70031A**

Questions:

1. What are the best features of this document?

2. How does this document meet your hardware and software development needs?

3. Do you find the organization of this data sheet easy to follow? If not, why?

4. What additions to the data sheet do you think would enhance the structure and subject?

5. What deletions from the data sheet could be made without affecting the overall usefulness?

6. Is there any incorrect or misleading information (what and where)?

7. How would you improve this document?

8. How would you improve our software, systems, and silicon products?

## rfPIC12C509AG/509AF PRODUCT IDENTIFICATION SYSTEM

To order or obtain information, e.g., on pricing or delivery, refer to the factory or the listed sales office.

**PART NO. X /XX XXX**

| | | | |
|---|---|---|---|
| **Pattern:** | Special Requirements | | |
| **Package:** | SO | = | 300 mil SOIC |
| | SS | = | 209 mil SSOP |
| | JW | = | 300 mil Windowed Ceramic Side Brazed |
| **Temperature Range:** | - | = | -40°C to +85°C |
| **Device** | rfPIC12C509AG<br>rfPIC12C509AF | | |

**Examples**

a) rfPIC12C509AG/JW Industrial Temp., Windowed CERDIP, 4 MHz

b) rfPIC12C509AG/SO Industrial Temp., SOIC package, 4 MHz

c) rfPIC12C509AF/SS Industrial Temp., SSOP package, 4 MHz

### Sales and Support

**Data Sheets**
Products supported by a preliminary Data Sheet may have an errata sheet describing minor operational differences and recommended workarounds. To determine if an errata sheet exists for a particular device, please contact one of the following:

1. Your local Microchip sales office
2. The Microchip Corporate Literature Center U.S. FAX: (480) 792-7277
3. The Microchip Worldwide Site (www.microchip.com)

Please specify which device, revision of silicon and Data Sheet (include Literature #) you are using.

**New Customer Notification System**
Register on our web site (www.microchip.com/cn) to receive the most current information on our products.

# WORLDWIDE SALES AND SERVICE

## AMERICAS

**Corporate Office**
2355 West Chandler Blvd.
Chandler, AZ 85224-6199
Tel: 480-792-7200 Fax: 480-792-7277
Technical Support: 480-792-7627
Web Address: http://www.microchip.com

**Rocky Mountain**
2355 West Chandler Blvd.
Chandler, AZ 85224-6199
Tel: 480-792-7966 Fax: 480-792-7456

**Atlanta**
500 Sugar Mill Road, Suite 200B
Atlanta, GA 30350
Tel: 770-640-0034 Fax: 770-640-0307

**Austin - Analog**
13740 North Highway 183
Building J, Suite 4
Austin, TX 78750
Tel: 512-257-3370 Fax: 512-257-8526

**Boston**
2 Lan Drive, Suite 120
Westford, MA 01886
Tel: 978-692-3848 Fax: 978-692-3821

**Boston - Analog**
Unit A-8-1 Millbrook Tarry Condominium
97 Lowell Road
Concord, MA 01742
Tel: 978-371-6400 Fax: 978-371-0050

**Chicago**
333 Pierce Road, Suite 180
Itasca, IL 60143
Tel: 630-285-0071 Fax: 630-285-0075

**Dallas**
4570 Westgrove Drive, Suite 160
Addison, TX 75001
Tel: 972-818-7423 Fax: 972-818-2924

**Dayton**
Two Prestige Place, Suite 130
Miamisburg, OH 45342
Tel: 937-291-1654 Fax: 937-291-9175

**Detroit**
Tri-Atria Office Building
32255 Northwestern Highway, Suite 190
Farmington Hills, MI 48334
Tel: 248-538-2250 Fax: 248-538-2260

**Los Angeles**
18201 Von Karman, Suite 1090
Irvine, CA 92612
Tel: 949-263-1888 Fax: 949-263-1338

**New York**
150 Motor Parkway, Suite 202
Hauppauge, NY 11788
Tel: 631-273-5305 Fax: 631-273-5335

**San Jose**
Microchip Technology Inc.
2107 North First Street, Suite 590
San Jose, CA 95131
Tel: 408-436-7950 Fax: 408-436-7955

**Toronto**
6285 Northam Drive, Suite 108
Mississauga, Ontario L4V 1X5, Canada
Tel: 905-673-0699 Fax: 905-673-6509

## ASIA/PACIFIC

**Australia**
Microchip Technology Australia Pty Ltd
Suite 22, 41 Rawson Street
Epping 2121, NSW
Australia
Tel: 61-2-9868-6733 Fax: 61-2-9868-6755

**China - Beijing**
Microchip Technology Consulting (Shanghai) Co., Ltd., Beijing Liaison Office
Unit 915
Bei Hai Wan Tai Bldg.
No. 6 Chaoyangmen Beidajie
Beijing, 100027, No. China
Tel: 86-10-85282100 Fax: 86-10-85282104

**China - Chengdu**
Microchip Technology Consulting (Shanghai) Co., Ltd., Chengdu Liaison Office
Rm. 2401, 24th Floor,
Ming Xing Financial Tower
No. 88 TIDU Street
Chengdu 610016, China
Tel: 86-28-6766200 Fax: 86-28-6766599

**China - Fuzhou**
Microchip Technology Consulting (Shanghai) Co., Ltd., Fuzhou Liaison Office
Rm. 531, North Building
Fujian Foreign Trade Center Hotel
73 Wusi Road
Fuzhou 350001, China
Tel: 86-591-7557563 Fax: 86-591-7557572

**China - Shanghai**
Microchip Technology Consulting (Shanghai) Co., Ltd.
Room 701, Bldg. B
Far East International Plaza
No. 317 Xian Xia Road
Shanghai, 200051
Tel: 86-21-6275-5700 Fax: 86-21-6275-5060

**China - Shenzhen**
Microchip Technology Consulting (Shanghai) Co., Ltd., Shenzhen Liaison Office
Rm. 1315, 13/F, Shenzhen Kerry Centre,
Renminnan Lu
Shenzhen 518001, China
Tel: 86-755-2350361 Fax: 86-755-2366086

**Hong Kong**
Microchip Technology Hongkong Ltd.
Unit 901-6, Tower 2, Metroplaza
223 Hing Fong Road
Kwai Fong, N.T., Hong Kong
Tel: 852-2401-1200 Fax: 852-2401-3431

**India**
Microchip Technology Inc.
India Liaison Office
Divyasree Chambers
1 Floor, Wing A (A3/A4)
No. 11, O'Shaugnessey Road
Bangalore, 560 025, India
Tel: 91-80-2290061 Fax: 91-80-2290062

**Japan**
Microchip Technology Japan K.K.
Benex S-1 6F
3-18-20, Shinyokohama
Kohoku-Ku, Yokohama-shi
Kanagawa, 222-0033, Japan
Tel: 81-45-471- 6166 Fax: 81-45-471-6122

**Korea**
Microchip Technology Korea
168-1, Youngbo Bldg. 3 Floor
Samsung-Dong, Kangnam-Ku
Seoul, Korea 135-882
Tel: 82-2-554-7200 Fax: 82-2-558-5934

**Singapore**
Microchip Technology Singapore Pte Ltd.
200 Middle Road
#07-02 Prime Centre
Singapore, 188980
Tel: 65-334-8870 Fax: 65-334-8850

**Taiwan**
Microchip Technology Taiwan
11F-3, No. 207
Tung Hua North Road
Taipei, 105, Taiwan
Tel: 886-2-2717-7175 Fax: 886-2-2545-0139

## EUROPE

**Denmark**
Microchip Technology Denmark ApS
Regus Business Centre
Lautrup hoj 1-3
Ballerup DK-2750 Denmark
Tel: 45 4420 9895 Fax: 45 4420 9910

**France**
Arizona Microchip Technology SARL
Parc d'Activite du Moulin de Massy
43 Rue du Saule Trapu
Batiment A - ler Etage
91300 Massy, France
Tel: 33-1-69-53-63-20 Fax: 33-1-69-30-90-79

**Germany**
Arizona Microchip Technology GmbH
Gustav-Heinemann Ring 125
D-81739 Munich, Germany
Tel: 49-89-627-144 0 Fax: 49-89-627-144-44

**Germany - Analog**
Lochhamer Strasse 13
D-82152 Martinsried, Germany
Tel: 49-89-895650-0 Fax: 49-89-895650-22

**Italy**
Arizona Microchip Technology SRL
Centro Direzionale Colleoni
Palazzo Taurus 1 V. Le Colleoni 1
20041 Agrate Brianza
Milan, Italy
Tel: 39-039-65791-1 Fax: 39-039-6899883

**United Kingdom**
Arizona Microchip Technology Ltd.
505 Eskdale Road
Winnersh Triangle
Wokingham
Berkshire, England RG41 5TU
Tel: 44 118 921 5869 Fax: 44-118 921-5820

08/01/01