



MICROCHIP

PIC12F752/HV752

Data Sheet

8-Pin, Flash-Based 8-Bit
CMOS Microcontrollers

**8-bit, 8-pin Devices Protected by Microchip's Low Pin Count Patent: U.S. Patent No. 5,847,450.
Additional U.S. and foreign patents and applications may be issued or pending.*

Note the following details of the code protection feature on Microchip devices:

- Microchip products meet the specification contained in their particular Microchip Data Sheet.
- Microchip believes that its family of products is one of the most secure families 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 Microchip products in a manner outside the operating specifications contained in Microchip's Data Sheets. Most likely, the person doing so is 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 products. Attempts to break Microchip's code protection feature may be a violation of the Digital Millennium Copyright Act. If such acts allow unauthorized access to your software or other copyrighted work, you may have a right to sue for relief under that Act.

Information contained in this publication regarding device applications and the like is provided only for your convenience and may be superseded by updates. It is your responsibility to ensure that your application meets with your specifications. MICROCHIP MAKES NO REPRESENTATIONS OR WARRANTIES OF ANY KIND WHETHER EXPRESS OR IMPLIED, WRITTEN OR ORAL, STATUTORY OR OTHERWISE, RELATED TO THE INFORMATION, INCLUDING BUT NOT LIMITED TO ITS CONDITION, QUALITY, PERFORMANCE, MERCHANTABILITY OR FITNESS FOR PURPOSE. Microchip disclaims all liability arising from this information and its use. Use of Microchip devices in life support and/or safety applications is entirely at the buyer's risk, and the buyer agrees to defend, indemnify and hold harmless Microchip from any and all damages, claims, suits, or expenses resulting from such use. No licenses are conveyed, implicitly or otherwise, under any Microchip intellectual property rights.

Trademarks

The Microchip name and logo, the Microchip logo, dsPIC, KEELOQ, KEELOQ logo, MPLAB, PIC, PICmicro, PICSTART, PIC³² logo, rPIC and UNI/O are registered trademarks of Microchip Technology Incorporated in the U.S.A. and other countries.


FilterLab, Hampshire, HI-TECH C, Linear Active Thermistor, MXDEV, MXLAB, SEEVAL and The Embedded Control Solutions Company are registered trademarks of Microchip Technology Incorporated in the U.S.A.

Analog-for-the-Digital Age, Application Maestro, chipKIT, chipKIT logo, CodeGuard, dsPICDEM, dsPICDEM.net, dsPICworks, dsSPEAK, ECAN, ECONOMONITOR, FanSense, HI-TIDE, In-Circuit Serial Programming, ICSP, Mindi, MiWi, MPASM, MPLAB Certified logo, MPLIB, MPLINK, mTouch, Omniscient Code Generation, PICC, PICC-18, PICDEM, PICDEM.net, PICkit, PICtail, REAL ICE, rLAB, Select Mode, Total Endurance, TSHARC, UniWinDriver, WiperLock and ZENA are trademarks of Microchip Technology Incorporated in the U.S.A. and other countries.

SQTP is a service mark of Microchip Technology Incorporated in the U.S.A.

All other trademarks mentioned herein are property of their respective companies.

© 2011, Microchip Technology Incorporated, Printed in the U.S.A., All Rights Reserved.

 Printed on recycled paper.

ISBN: 978-1-61341-798-0

Microchip received ISO/TS-16949:2009 certification for its worldwide headquarters, design and wafer fabrication facilities in Chandler and Tempe, Arizona; Gresham, Oregon and design centers in California and India. The Company's quality system processes and procedures are for its PIC[®] MCUs and dsPIC[®] DSCs, KEELOQ[®] code hopping devices, Serial EEPROMs, microperipherals, nonvolatile memory and analog products. In addition, Microchip's quality system for the design and manufacture of development systems is ISO 9001:2000 certified.

QUALITY MANAGEMENT SYSTEM
CERTIFIED BY DNV
== ISO/TS 16949:2009 ==

8-Pin Flash-Based, 8-Bit CMOS Microcontrollers

High-Performance RISC CPU:

- Only 35 Instructions to Learn:
 - All single-cycle instructions except branches
- Operating Speed:
 - DC – 20 MHz clock input
 - DC – 200 ns instruction cycle
- 1024 x 14 On-chip Flash Program Memory
- Self Read/Write Program Memory
- 64 x 8 General Purpose Registers (SRAM)
- Interrupt Capability
- 8-Level Deep Hardware Stack
- Direct, Indirect and Relative Addressing modes

Microcontroller Features:

- Precision Internal Oscillator:
 - Factory calibrated to $\pm 1\%$, typical
 - Software selectable frequency:
 - 8 MHz, 4 MHz, 1 MHz or 31 kHz
 - Software tunable
- Power-Saving Sleep mode
- Voltage Range (PIC12F752):
 - 2.0V to 5.5V
- Shunt Voltage Regulator (PIC12HV752)
 - 2.0V to user defined
 - 5 volt regulation
 - 4 mA to 50 mA shunt range
- Multiplexed Master Clear with Pull-up/Input Pin
- Interrupt-on-Change Pins
- Individually Programmable Weak Pull-ups
- Power-on Reset (POR)
- Power-up Timer (PWRT)
- Brown-out Reset (BOR)
- Watchdog Timer (WDT) with Internal Oscillator for Reliable Operation
- Industrial and Extended Temperature Range
- High Endurance Flash:
 - 100,000 write Flash endurance
 - Flash retention: >40 years
- Programmable Code Protection
- In-Circuit Debug (ICD) via Two Pins
- In-Circuit Serial Programming™ (ICSP™) via Two Pins

Low-Power Features:

- Standby Current:
 - 50 nA @ 2.0V, typical
- Operating Current:
 - 11 μ A @ 32 kHz, 2.0V, typical
 - 260 μ A @ 4 MHz, 2.0V, typical
- Watchdog Timer Current:
 - <1 μ A @ 2.0V, typical

Peripheral Features:

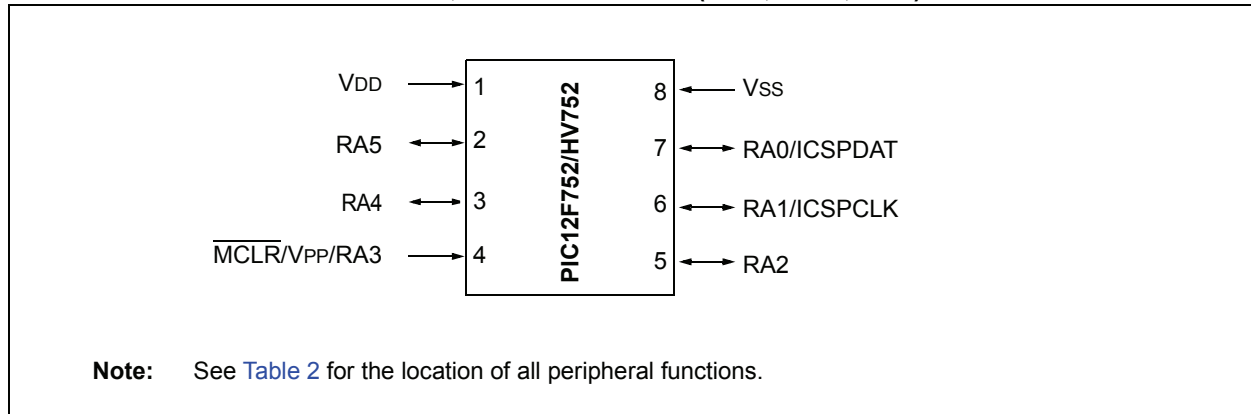
- 5 I/O Pins and 1 Input-only Pin
- High Current Source/Sink:
 - 50 mA I/O, (2 pins)
 - 25 mA I/O, (4 pins)
- 2 High-Speed Analog Comparator modules:
 - 20 ns response time
 - Fixed Voltage Reference (FVR)
 - Programmable on-chip voltage reference via integrated 5-bit DAC
 - Internal/external inputs and outputs (selectable)
 - Built-in Hysteresis (software selectable)
- A/D Converter:
 - 10-bit resolution
 - 4 external channels
 - 2 internal reference voltage channels
- Dual Range Digital-to-Analog Converter (DAC):
 - 5-bit resolution
 - Full Range or Limited Range output
 - 4 mV steps @ 2.0V (Limited Range)
 - 65 mV steps @ 2.0V (Full Range)
- Fixed Voltage Reference (FVR), 1.2V reference
- Capture, Compare, PWM (CCP) module:
 - 16-bit Capture, max. resolution = 12.5 ns
 - Compare, max. resolution = 200 ns
 - 10-bit PWM, max. frequency = 20 kHz
- Timer0: 8-Bit Timer/Counter with 8-Bit Prescaler
- Enhanced Timer1:
 - 16-bit Timer/Counter with Prescaler
 - External Timer1 Gate (count enable)
 - 4 Selectable Clock sources
- Timer2: 8-Bit Timer/Counter with Prescaler
 - 8-Bit Period Register and Postscaler
- Hardware Limit Timer (HLT):
 - 8-bit Timer with Prescaler
 - 8-bit period register and postscaler
 - Asynchronous H/W Reset sources
- Complementary Output Generator (COG):
 - Complementary Waveforms from selectable sources
 - 2 I/O (50 mA) for direct MOSFET drive
 - Rising and/or Falling edge dead-band control
 - Phase control, Blanking control
 - Auto-shutdown

PIC12F752/HV752

TABLE 1: PIC12F752/HV752 FEATURE SUMMARY

| Device | Flash Program Memory (User) (words) | Self Read/Write Flash Memory | SRAM (bytes) | I/Os | 10-bit A/D (ch) | Comparators | Timers 8/16-bit | CCP | Complementary Output Generator (COG) | Shunt Regulator |
|------------|-------------------------------------|------------------------------|--------------|------|-----------------|-------------|-----------------|-----|--------------------------------------|-----------------|
| PIC12F752 | 1024 | Y | 64 | 6 | 4 | 2 | 3/1 | 1 | Y | N |
| PIC12HV752 | 1024 | Y | 64 | 6 | 4 | 2 | 3/1 | 1 | Y | Y |

FIGURE 1: 8-PIN DIAGRAM, PIC12F752/HV752 (PDIP, SOIC, DFN)



PIC12F752/HV752

TABLE 2: PIC12F752/HV752 PIN SUMMARY (PDIP, SOIC, DFN)

| I/O | Pin | ADC | Comparators | Timers | CCP | Interrupts | Pull-up | Complementary Output Generator (COG) | Voltage Reference | Basic |
|--------------------|-----|-----|------------------|--------------------|------|------------|------------------|---|-------------------|----------|
| RA0 ⁽⁵⁾ | 7 | AN0 | C1IN0+ C2IN0+ | — | — | IOC | Y | COG1OUT1 ⁽²⁾ | DACOUT REFOUT | ICSPDAT |
| RA1 | 6 | AN1 | C1IN0- C2IN0- | — | — | IOC | Y | — | VREF+ | ICSPCLK |
| RA2 ⁽⁵⁾ | 5 | AN2 | C1OUT C2OUT | T0CKI | CCP1 | IOC INT | Y | COG1OUT0 ⁽²⁾ | — | — |
| RA3 ⁽¹⁾ | 4 | — | — | T1G ⁽³⁾ | — | IOC | Y ⁽⁴⁾ | COG1FLT ⁽³⁾ | — | MCLR/VPP |
| RA4 | 3 | AN3 | C1IN1- | T1G ⁽²⁾ | — | IOC | Y | COG1FLT ⁽²⁾ COG1OUT1 ⁽³⁾ | — | CLKOUT |
| RA5 | 2 | — | C2IN1- | T1CKI | — | IOC | Y | COG1OUT0 ⁽³⁾ | — | CLKIN |
| — | 1 | — | — | — | — | — | — | — | — | VDD |
| — | 8 | — | — | — | — | — | — | — | — | VSS |

* Alternate pin function.

- Note 1:** Input only.
- 2:** Default pin function via the APFCON register.
- 3:** Alternate pin function via the APFCON register.
- 4:** RA3 pull-up is enabled when pin is configured as MCLR in Configuration Word.
- 5:** The port pins for the primary COG1OUTx pins have High Power (HP) output drivers.

PIC12F752/HV752

Table of Contents

| | | |
|------|---|-----|
| 1.0 | Device Overview | 9 |
| 2.0 | Memory Organization | 11 |
| 3.0 | Flash Program Memory Self Read/Self Write Control | 29 |
| 4.0 | Oscillator Module | 39 |
| 5.0 | I/O Ports | 45 |
| 6.0 | Timer0 Module | 55 |
| 7.0 | Timer1 Module with Gate Control | 57 |
| 8.0 | Timer2 Module | 69 |
| 9.0 | Hardware Limit Timer (HLT) Module | 73 |
| 10.0 | Capture/Compare/PWM Modules | 77 |
| 11.0 | Complementary Output Generator (COG) Module | 85 |
| 12.0 | Analog-to-Digital Converter (ADC) Module | 101 |
| 13.0 | Fixed Voltage Reference (FVR) | 113 |
| 14.0 | Digital-to-Analog Converter (DAC) Module | 115 |
| 15.0 | Comparator Module | 121 |
| 16.0 | Instruction Set Summary | 131 |
| 17.0 | Special Features of the CPU | 141 |
| 18.0 | Shunt Regulator (PIC12HV752 Only) | 161 |
| 19.0 | Development Support | 163 |
| 20.0 | Electrical Specifications | 167 |
| 21.0 | DC and AC Characteristics Graphs and Charts | 191 |
| 22.0 | Packaging Information | 193 |
| | Index | 207 |
| | The Microchip Web Site | 207 |
| | Customer Change Notification Service | 207 |
| | Customer Support | 207 |
| | Reader Response | 208 |
| | Product Identification System | 209 |

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@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)

When contacting a sales office, 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 to receive the most current information on all of our products.

PIC12F752/HV752

NOTES:

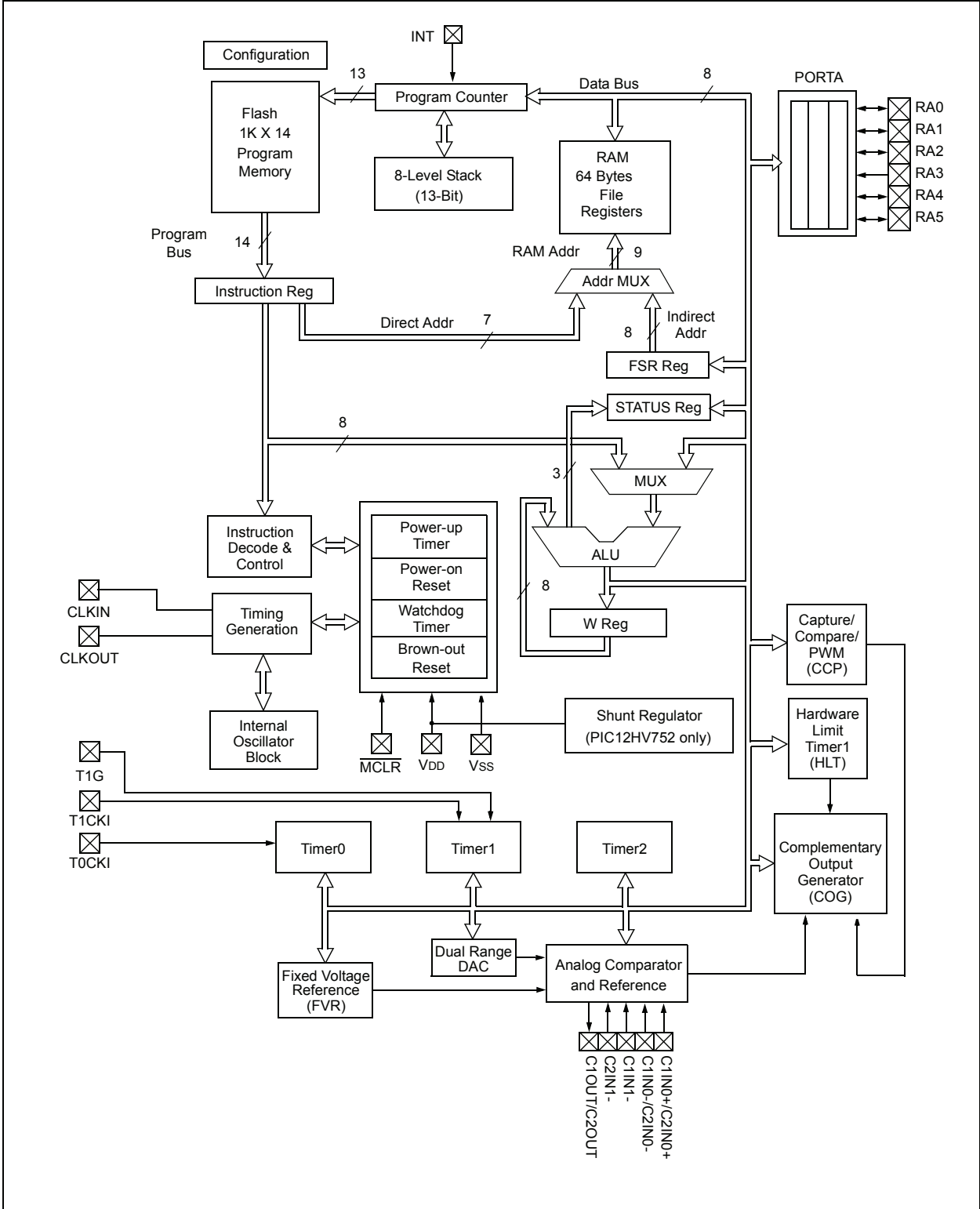
PIC12F752/HV752

1.0 DEVICE OVERVIEW

Block Diagrams and pinout descriptions of the devices are in [Figure 1-1](#) and [Table 1-1](#).

The PIC12F752/HV752 devices are covered by this data sheet. They are available in 8-pin PDIP, SOIC and DFN packages.

FIGURE 1-1: PIC12F752/HV752 BLOCK DIAGRAM



PIC12F752/HV752

TABLE 1-1: PIC12F752/HV752 PINOUT DESCRIPTION

| Name | Function | Input Type | Output Type | Description |
|--|----------|------------|-------------|--|
| RA0/COG1OUT1 ⁽²⁾ /C1IN0+/ C2IN0+/AN0/DACOUT/ REFOUT/ ICSPDAT | RA0 | TTL | HP | General purpose I/O with IOC and WPU. |
| | COG1OUT1 | — | HP | COG output channel 1. |
| | C1IN0+ | AN | — | Comparator C1 positive input. |
| | C2IN0+ | AN | — | Comparator C2 positive input. |
| | AN0 | AN | — | A/D Channel 0 input. |
| | DACOUT | — | AN | DAC unbuffered Voltage Reference output. |
| | REFOUT | — | AN | DAC/FVR buffered Voltage Reference output. |
| RA1/C1IN0-/C2IN0-/AN1/ VREF+/ICSPCLK | RA1 | TTL | CMOS | General purpose I/O with IOC and WPU. |
| | C1IN0- | AN | — | Comparator C1 negative input. |
| | C2IN0- | AN | — | Comparator C2 negative input. |
| | AN1 | AN | — | A/D Channel 1 input. |
| | VREF+ | AN | — | A/D Positive Voltage Reference input. |
| RA2/INT/CCP1/C2OUT/ C1OUT/T0CKI/ COG1OUT0 ⁽²⁾ /AN2 | RA2 | ST | HP | General purpose I/O with IOC and WPU. |
| | INT | ST | — | External interrupt. |
| | CCP1 | ST | HP | Capture/Compare/PWM 1. |
| | C2OUT | — | HP | Comparator C2 output. |
| | C1OUT | — | HP | Comparator C1 output. |
| | T0CKI | ST | — | Timer0 clock input. |
| | COG1OUT0 | — | HP | COG output channel 0. |
| RA3 ⁽¹⁾ /T1G ⁽³⁾ /COG1FLT ⁽³⁾ / VPP/MCLR ⁽⁴⁾ | RA3 | TTL | — | General purpose input with WPU. |
| | T1G | ST | — | Timer1 Gate input. |
| | COG1FLT | ST | — | COG auto-shutdown fault input. |
| | VPP | HV | — | Programming voltage. |
| | MCLR | ST | — | Master Clear w/internal pull-up. |
| RA4/T1G ⁽²⁾ /COG1OUT1 ⁽³⁾ / COG1FLT ⁽²⁾ /C1IN1-/AN3/ CLKOUT | RA4 | TTL | CMOS | General purpose I/O with IOC and WPU. |
| | T1G | ST | — | Timer1 Gate input. |
| | COG1OUT1 | — | CMOS | COG output channel 1 |
| | COG1FLT | ST | — | COG auto-shutdown fault input. |
| | C1IN1- | AN | — | Comparator C1 negative input. |
| | AN3 | AN | — | A/D Channel 3 input. |
| RA5/T1CKI/COG1OUT0 ⁽³⁾ / C2IN1-/CLKIN | RA5 | TTL | CMOS | General purpose I/O with IOC and WPU. |
| | T1CKI | ST | — | Timer1 clock input. |
| | COG1OUT0 | — | CMOS | COG output channel 0. |
| | C2IN1- | AN | — | Comparator C2 negative input. |
| | CLKIN | ST | — | External Clock input (EC mode). |
| VDD | VDD | Power | — | Positive supply. |
| VSS | VSS | Power | — | Ground reference. |

Legend: AN = Analog input or output CMOS = CMOS compatible input or output
TTL = TTL compatible input ST = Schmitt Trigger input with CMOS levels
HP = High Power HV = High Voltage

* Alternate pin function.

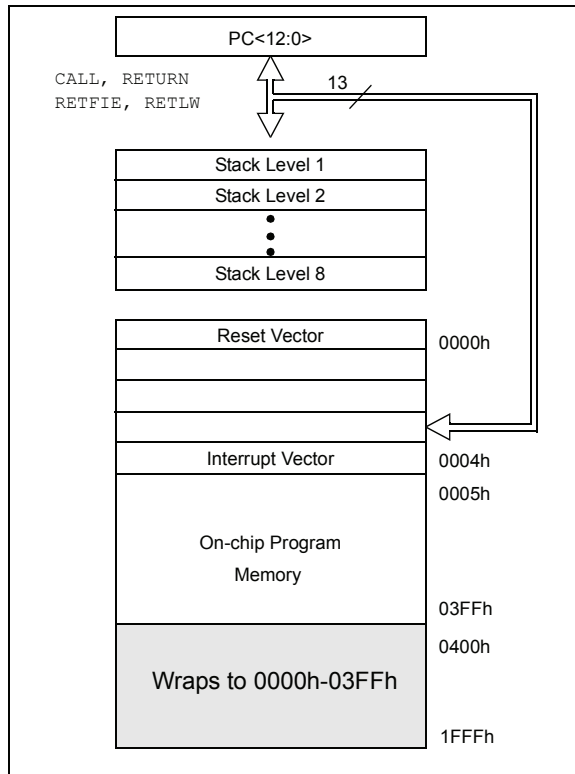
- Note** 1: Input only.
2: Default pin function via the APFCON register.
3: Alternate pin function via the APFCON register.
4: RA3 pull-up is enabled when pin is configured as MCLR in Configuration Word.

2.0 MEMORY ORGANIZATION

2.1 Program Memory Organization

The PIC12F752/HV752 has a 13-bit program counter capable of addressing an 8K x 14 program memory space. Only the first 1K x 14 (0000h-03FFh) is physically implemented. Accessing a location above these boundaries will cause a wrap-around within the first 1K x 14 space for PIC12F752/HV752. The Reset vector is at 0000h and the interrupt vector is at 0004h (see Figure 2-1).

FIGURE 2-1: PROGRAM MEMORY MAP AND STACK FOR THE PIC12F752/HV752



2.2 Data Memory Organization

The data memory (see Figure 2-2) is partitioned into four banks, which contain the General Purpose Registers (GPR) and the Special Function Registers (SFR). The Special Function Registers are located in the first 32 locations of each bank. Register locations 40h-6Fh in Bank 0 are General Purpose Registers, implemented as static RAM. Register locations 70h-7Fh in Bank 0 are Common RAM and shared as the last 16 addresses in all Banks. All other RAM is unimplemented and returns '0' when read. The RP<1:0> bits of the STATUS register are the bank select bits.

| RP1 | RP0 | |
|-----|-----|----------------------|
| 0 | 0 | → Bank 0 is selected |
| 0 | 1 | → Bank 1 is selected |
| 1 | 0 | → Bank 2 is selected |
| 1 | 1 | → Bank 3 is selected |

PIC12F752/HV752

2.2.1 GENERAL PURPOSE REGISTER FILE

The register file is organized as 64 x 8 in the PIC12F752/HV752. Each register is accessed, either directly or indirectly, through the File Select Register (FSR) (see [Section 2.4 “Indirect Addressing, INDF and FSR Registers”](#)).

2.2.2 SPECIAL FUNCTION REGISTERS

The Special Function Registers are registers used by the CPU and peripheral functions for controlling the desired operation of the device (see [Table 2-1](#)). These registers are static RAM.

The special registers can be classified into two sets: core and peripheral. The Special Function Registers associated with the “core” are described in this section. Those related to the operation of the peripheral features are described in the section of that peripheral feature.

FIGURE 2-2: DATA MEMORY MAP OF THE PIC12F752/HV752

| BANK 0 | | BANK 1 | | BANK 2 | | BANK 3 | |
|--------------------------|-----|-------------------------------|-----|-------------------------------|------|-------------------------------|------|
| INDF | 00h | INDF | 80h | INDF | 100h | INDF | 180h |
| TMR0 | 01h | OPTION_REG | 81h | TMR0 | 101h | OPTION_REG | 181h |
| PCL | 02h | PCL | 82h | PCL | 102h | PCL | 182h |
| STATUS | 03h | STATUS | 83h | STATUS | 103h | STATUS | 183h |
| FSR | 04h | FSR | 84h | FSR | 104h | FSR | 184h |
| PORTA | 05h | TRISA | 85h | LATA | 105h | ANSELA | 185h |
| — | 06h | — | 86h | — | 106h | — | 186h |
| — | 07h | — | 87h | — | 107h | — | 187h |
| IOCAF | 08h | IOCAP | 88h | IOCAN | 108h | APFCON | 188h |
| — | 09h | — | 89h | — | 109h | OSCTUNE | 189h |
| PCLATH | 0Ah | PCLATH | 8Ah | PCLATH | 10Ah | PCLATH | 18Ah |
| INTCON | 0Bh | INTCON | 8Bh | INTCON | 10Bh | INTCON | 18Bh |
| PIR1 | 0Ch | PIE1 | 8Ch | WPUA | 10Ch | PMCON1 | 18Ch |
| PIR2 | 0Dh | PIE2 | 8Dh | SLRCONA | 10Dh | PMCON2 | 18Dh |
| — | 0Eh | — | 8Eh | — | 10Eh | PMADRL | 18Eh |
| TMR1L | 0Fh | OSCCON | 8Fh | PCON | 10Fh | PMADRH | 18Fh |
| TMR1H | 10h | FVRCON | 90h | TMR2 | 110h | PMDATL | 190h |
| T1CON | 11h | DACCON0 | 91h | PR2 | 111h | PMDATH | 191h |
| T1GCON | 12h | DACCON1 | 92h | T2CON | 112h | COG1PH | 192h |
| CCPR1L | 13h | — | 93h | HLTMR1 | 113h | COG1BLK | 193h |
| CCPR1H | 14h | — | 94h | HLTPR1 | 114h | COG1DB | 194h |
| CCP1CON | 15h | — | 95h | HLT1CON0 | 115h | COG1CON0 | 195h |
| — | 16h | — | 96h | HLT1CON1 | 116h | COG1CON1 | 196h |
| — | 17h | — | 97h | — | 117h | COG1ASD | 197h |
| — | 18h | — | 98h | — | 118h | — | 198h |
| — | 19h | — | 99h | — | 119h | — | 199h |
| — | 1Ah | — | 9Ah | — | 11Ah | — | 19Ah |
| — | 1Bh | CM2CON0 | 9Bh | — | 11Bh | — | 19Bh |
| ADRESL | 1Ch | CM2CON1 | 9Ch | — | 11Ch | — | 19Ch |
| ADRESH | 1Dh | CM1CON0 | 9Dh | — | 11Dh | — | 19Dh |
| ADCON0 | 1Eh | CM1CON1 | 9Eh | — | 11Eh | — | 19Eh |
| ADCON1 | 1Fh | CMOUT | 9Fh | — | 11Fh | — | 19Fh |
| Unimplemented | 20h | | A0h | | 120h | | 1A0h |
| Unimplemented | 3Fh | | | | | | |
| General Purpose Register | 40h | Unimplemented | | Unimplemented | | Unimplemented | |
| 48 Bytes | 6Fh | | EFh | | 16Fh | | 1EFh |
| Common RAM | 70h | Common RAM (Accesses 70h-7Fh) | F0h | Common RAM (Accesses 70h-7Fh) | 170h | Common RAM (Accesses 70h-7Fh) | 1F0h |
| 16 Bytes | 7Fh | | FFh | | 17Fh | | 1FFh |

Legend: ■ = Unimplemented data memory locations, read as '0'.

PIC12F752/HV752

TABLE 2-1: PIC12F752/HV752 SPECIAL REGISTERS SUMMARY BANK 0

| Adr | Name | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | Value on POR/BOR Reset | Value on all other Resets ⁽¹⁾ | |
|---------------|---------|--|-----------|-------------|--|-----------------|----------------------|------------|----------------------|------------------------|--|------|
| Bank 0 | | | | | | | | | | | | |
| 00h | INDF | Addressing this location uses contents of FSR to address data memory (not a physical register) | | | | | | | | xxxx xxxx | xxxx xxxx | |
| 01h | TMR0 | Holding register for the 8-bit TMR0 | | | | | | | | xxxx xxxx | uuuu uuuu | |
| 02h | PCL | Program Counter's (PC) Least Significant Byte | | | | | | | | 0000 0000 | 0000 0000 | |
| 03h | STATUS | IRP | RP1 | RP0 | \overline{TO} | \overline{PD} | Z | DC | C | 0001 1xxx | 000q quuu | |
| 04h | FSR | Indirect Data Memory Address Pointer | | | | | | | | xxxx xxxx | uuuu uuuu | |
| 05h | PORTA | — | — | RA5 | RA4 | RA3 | RA2 | RA1 | RA0 | --xx xxxx | --uu uuuu | |
| 06h | — | Unimplemented | | | | | | | | — | — | |
| 07h | — | Unimplemented | | | | | | | | — | — | |
| 08h | IOCAF | — | — | IOCAF5 | IOCAF4 | IOCAF3 | IOCAF2 | IOCAF1 | IOCAF0 | --00 0000 | --00 0000 | |
| 09h | — | Unimplemented | | | | | | | | — | — | |
| 0Ah | PCLATH | — | — | — | Write buffer for upper 5 bits of program counter | | | | --- | 0000 | --- | 0000 |
| 0Bh | INTCON | GIE | PEIE | T0IE | INTE | IOCIE | T0IF | INTF | IOCIF ⁽²⁾ | 0000 0000 | 0000 0000 | |
| 0Ch | PIR1 | TMR1GIF | ADIF | — | — | — | HLTMR1IF | TMR2IF | TMR1IF | 00---000 | 00---000 | |
| 0Dh | PIR2 | — | — | C2IF | C1IF | — | COG1IF | — | CCP1IF | --00 -0-0 | --00 -0-0 | |
| 0Eh | — | Unimplemented | | | | | | | | — | — | |
| 0Fh | TMR1L | Holding register for the Least Significant Byte of the 16-bit TMR1 | | | | | | | | xxxx xxxx | uuuu uuuu | |
| 10h | TMR1H | Holding register for the Most Significant Byte of the 16-bit TMR1 | | | | | | | | xxxx xxxx | uuuu uuuu | |
| 11h | T1CON | TMR1CS<1:0> | | T1CKPS<1:0> | | Reserved | $\overline{T1SYNC}$ | — | TMR1ON | 0000 00-0 | uuuu uu-u | |
| 12h | T1GCON | TMR1GE | T1GPOL | T1GTM | T1GSPM | T1GGO/DONE | T1GVAL | T1GSS<1:0> | | 0000 0x00 | uuuu uxuu | |
| 13h | CCPR1L | Capture/Compare/PWM Register1 Low Byte | | | | | | | | xxxx xxxx | uuuu uuuu | |
| 14h | CCPR1H | Capture/Compare/PWM Register1 High Byte | | | | | | | | xxxx xxxx | uuuu uuuu | |
| 15h | CCP1CON | — | — | DC1B<1:0> | | CCP1M<3:0> | | | — | --00 0000 | --00 0000 | |
| 16h to 1Bh | — | Unimplemented | | | | | | | | — | — | |
| 1Ch | ADRESL | Least Significant 2 bits of the left shifted result or 8 bits of the right shifted result | | | | | | | | xxxx xxxx | uuuu uuuu | |
| 1Dh | ADRESH | Most Significant 8 bits of the left shifted A/D result or 2 bits of right shifted result | | | | | | | | xxxx xxxx | uuuu uuuu | |
| 1Eh | ADCON0 | ADFM | VCFG | CHS<3:0> | | | $\overline{GO/DONE}$ | ADON | — | 0000 0000 | 0000 0000 | |
| 1Fh | ADCON1 | — | ADCS<2:0> | | | — | — | — | — | -000 ---- | -000 ---- | |

Legend: — = Unimplemented locations read as '0', u = unchanged, x = unknown, q = value depends on condition shaded = unimplemented

Note 1: Other (non Power-up) Resets include MCLR Reset and Watchdog Timer Reset during normal operation.

Note 2: MCLR and WDT Reset does not affect the previous value data latch. The IOCIF bit will be cleared upon Reset but will set again if the mismatch exists.

PIC12F752/HV752

TABLE 2-2: PIC12F752/HV752 SPECIAL REGISTERS SUMMARY BANK 1

| Addr | Name | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | Value on POR/BOR Reset | Values on all other Resets ⁽¹⁾ | | |
|---------------|------------|--|--------|------------|--|-----------------------|----------|--------|-----------------------|------------------------|---|------|------|
| Bank 1 | | | | | | | | | | | | | |
| 80h | INDF | Addressing this location uses contents of FSR to address data memory (not a physical register) | | | | | | | | xxxx xxxx | uuuu uuuu | | |
| 81h | OPTION_REG | RAPU | INTEDG | T0CS | T0SE | PSA | PS<2:0> | | | 1111 1111 | 1111 1111 | | |
| 82h | PCL | Program Counter's (PC) Least Significant Byte | | | | | | | | 0000 0000 | 0000 0000 | | |
| 83h | STATUS | IRP | RP1 | RP0 | \overline{TO} | \overline{PD} | Z | DC | C | 0001 1xxx | 000q quuu | | |
| 84h | FSR | Indirect Data Memory Address Pointer | | | | | | | | xxxx xxxx | uuuu uuuu | | |
| 85h | TRISA | — | — | TRISA5 | TRISA4 | TRISA3 ⁽³⁾ | TRISA2 | TRISA1 | TRISA0 | --11 1111 | --11 1111 | | |
| 86h | — | Unimplemented | | | | | | | | — | — | | |
| 87h | — | Unimplemented | | | | | | | | — | — | | |
| 88h | IOCAP | — | — | IOCAP5 | IOCAP4 | IOCAP3 | IOCAP2 | IOCAP1 | IOCAP0 | --00 0000 | --00 0000 | | |
| 89h | — | Unimplemented | | | | | | | | — | — | | |
| 8Ah | PCLATH | — | — | — | Write buffer for upper 5 bits of program counter | | | | | --- | 0000 | --- | 0000 |
| 8Bh | INTCON | GIE | PEIE | TOIE | INTE | IOCFIE | TOIF | INTF | IOCFIF ⁽²⁾ | 0000 0000 | 0000 0000 | | |
| 8Ch | PIE1 | TMR1GIE | ADIE | — | — | — | HLTMR1IE | TMR2IE | TMR1IE | 00-- -000 | 00-- -000 | | |
| 8Dh | PIE2 | — | — | C2IE | C1IE | — | COG1IE | — | CCP1IE | --00 -0-0 | --00 -0-0 | | |
| 8Eh | — | Unimplemented | | | | | | | | — | — | | |
| 8Fh | OSCCON | — | — | IRCF<1:0> | | — | HTS | LTS | — | --01 -00- | --uu -uu- | | |
| 90h | FVRCON | FVREN | FVRRDY | FVR-BUFEN | FVR-BUFSS | — | — | — | — | 0000 ---- | 0000 ---- | | |
| 91h | DACCON0 | DACEN | DACRNG | DACOE | — | — | DACPSS0 | — | — | 000- -0-- | 000- -0-- | | |
| 92h | DACCON1 | — | — | — | DACR<4:0> | | | | --- | 0000 | --- | 0000 | |
| 93h to 9Ah | — | Unimplemented | | | | | | | | — | — | | |
| 9Bh | CM2CON0 | C2ON | C2OUT | C2OE | C2POL | C2ZLF | C2SP | C2HYS | C2SYNC | 0000 0100 | 0000 0100 | | |
| 9Ch | CM2CON1 | C2INTP | C2INTN | C2PCH<1:0> | | — | — | — | C2NCH0 | 0000 ---0 | 0000 ---0 | | |
| 9Dh | CM1CON0 | C1ON | C1OUT | C1OE | C1POL | C1ZLF | C1SP | C1HYS | C1SYNC | 0000 0100 | 0000 0100 | | |
| 9Eh | CM1CON1 | C1INTP | C1INTN | C1PCH<1:0> | | — | — | — | C1NCH0 | 0000 ---0 | 0000 ---0 | | |
| 9Fh | CMOUT | — | — | — | — | — | — | MC2OUT | MC1OUT | ---- --00 | ---- --00 | | |

Legend: — = Unimplemented locations read as '0', u = unchanged, x = unknown, q = value depends on condition shaded = unimplemented

Note 1: Other (non Power-up) Resets include MCLR Reset and Watchdog Timer Reset during normal operation.

Note 2: MCLR and WDT Reset does not affect the previous value data latch. The IOCFIE bit will be cleared upon Reset but will set again if the mismatch exists.

Note 3: TRISA3 always reads '1'.

PIC12F752/HV752

TABLE 2-3: PIC12F752/HV752 SPECIAL REGISTERS SUMMARY BANK 2

| Adr | Name | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | Value on POR/BOR Reset | Value on all other Resets ⁽¹⁾ | |
|---------------|----------|--|--------------|--------|--|-----------------|--------|------------------|----------------------|------------------------|--|-----------|
| Bank 2 | | | | | | | | | | | | |
| 100h | INDF | Addressing this location uses contents of FSR to address data memory (not a physical register) | | | | | | | | xxxx xxxx | xxxx xxxx | |
| 101h | TMR0 | Holding Register for the 8-bit Timer0 Register | | | | | | | | xxxx xxxx | uuuu uuuu | |
| 102h | PCL | Program Counter's (PC) Least Significant Byte | | | | | | | | 0000 0000 | 0000 0000 | |
| 103h | STATUS | IRP | RP1 | RP0 | \overline{TO} | \overline{PD} | Z | DC | C | 0001 1xxx | 000q quuu | |
| 104h | FSR | Indirect Data Memory Address Pointer | | | | | | | | xxxx xxxx | uuuu uuuu | |
| 105h | LATA | — | — | LATA5 | LATA4 | — | LATA2 | LATA1 | LATA0 | --xx -xxx | --uu -uuu | |
| 106h | — | Unimplemented | | | | | | | | — | — | |
| 107h | — | Unimplemented | | | | | | | | — | — | |
| 108h | IOCAN | — | — | IOCAN5 | IOCAN4 | IOCAN3 | IOCAN2 | IOCAN1 | IOCAN0 | --00 0000 | --00 0000 | |
| 109h | — | Unimplemented | | | | | | | | — | — | |
| 10Ah | PCLATH | — | — | — | Write buffer for upper 5 bits of program counter | | | | | ---0 0000 | ---0 0000 | |
| 10Bh | INTCON | GIE | PEIE | T0IE | INTE | IOCIE | T0IF | INTF | IOCIF ⁽²⁾ | 0000 0000 | 0000 0000 | |
| 10Ch | WPUA | — | — | WPUA5 | WPUA4 | WPUA3 | WPUA2 | WPUA1 | WPUA0 | --00 0000 | --00 0000 | |
| 10Dh | SLRCONA | — | — | — | — | — | SLRA2 | — | SLRA0 | ---- -0-0 | ---- -0-0 | |
| 10Eh | — | Unimplemented | | | | | | | | — | — | |
| 10Fh | PCON | — | — | — | — | — | — | \overline{POR} | \overline{BOR} | ---- --qq | ---- --uu | |
| 110h | TMR2 | Holding Register for the 8-bit Timer2 Register | | | | | | | | 0000 0000 | 0000 0000 | |
| 111h | PR2 | Timer2 Period Register | | | | | | | | 1111 1111 | 1111 1111 | |
| 112h | T2CON | — | TOUTPS<3:0> | | | | TMR2ON | T2CKPS<1:0> | | | -000 0000 | -000 0000 |
| 113h | HLTMR1 | Holding Register for the 8-bit Hardware Limit Timer1 Register | | | | | | | | 0000 0000 | 0000 0000 | |
| 114h | HLTPR1 | Hardware Limit Timer1 Period Register | | | | | | | | 1111 1111 | 1111 1111 | |
| 115h | HLT1CON0 | — | H1OUTPS<3:0> | | | | H1ON | H1CKPS<1:0> | | | -000 0000 | -000 0000 |
| 116h | HLT1CON1 | — | — | — | H1ERS<2:0> | | | H1FEREN | H1REREN | ---0 0000 | ---0 0000 | |
| 117h to 11Fh | — | Unimplemented | | | | | | | | — | — | |

Legend: — = Unimplemented locations read as '0', u = unchanged, x = unknown, q = value depends on condition shaded = unimplemented

Note 1: Other (non Power-up) Resets include MCLR Reset and Watchdog Timer Reset during normal operation.

Note 2: MCLR and WDT Reset does not affect the previous value data latch. The IOCIF bit will be cleared upon Reset but will set again if the mismatch exists.

PIC12F752/HV752

TABLE 2-4: PIC12F752/HV752 SPECIAL FUNCTION REGISTERS SUMMARY BANK 3

| Addr | Name | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | Value on POR/BOR Reset | Values on all other Resets ⁽¹⁾ | | |
|---------------|------------|--|---------|--|--|-----------------|-------------|-----------|----------------------|------------------------|---|-----------|-----------|
| Bank 3 | | | | | | | | | | | | | |
| 180h | INDF | Addressing this location uses contents of FSR to address data memory (not a physical register) | | | | | | | | xxxx xxxx | uuuu uuuu | | |
| 181h | OPTION_REG | RAPU | INTEDG | T0CS | T0SE | PSA | PS<2:0> | | | 1111 1111 | 1111 1111 | | |
| 182h | PCL | Program Counter's (PC) Least Significant Byte | | | | | | | | | 0000 0000 | 0000 0000 | |
| 183h | STATUS | IRP | RP1 | RP0 | \overline{TO} | \overline{PD} | Z | DC | C | 0001 1xxx | 000q quuu | | |
| 184h | FSR | Indirect Data Memory Address Pointer | | | | | | | | | xxxx xxxx | uuuu uuuu | |
| 185h | ANSELA | — | — | ANSA5 | ANSA4 | — | ANSA2 | ANSA1 | ANSA0 | --11 -111 | --11 -111 | | |
| 186h | — | Unimplemented | | | | | | | | | — | — | |
| 187h | — | Unimplemented | | | | | | | | | — | — | |
| 188h | APFCON | — | — | — | T1GSEL | — | COG1FSEL | COG1O1SEL | COG1O0SEL | ---0 -000 | ---0 -000 | | |
| 189h | OSCTUNE | — | — | — | TUN<4:0> | | | | --- | 0000 | ---u uuuu | | |
| 18Ah | PCLATH | — | — | — | Write buffer for upper 5 bits of program counter | | | | --- | 0000 | ---0 0000 | | |
| 18Bh | INTCON | GIE | PEIE | T0IE | INTE | IOCIE | T0IF | INTF | IOCIF ⁽²⁾ | 0000 0000 | 0000 0000 | | |
| 18Ch | PMCON1 | — | — | — | — | — | WREN | WR | RD | ---- -000 | ---- -000 | | |
| 18Dh | PMCON2 | Program Memory Control Register 2 (not a physical register) | | | | | | | | | ---- | ---- | |
| 18Eh | PMADRL | Program Memory Address Register Low Byte | | | | | | | | | 0000 0000 | 0000 0000 | |
| 18Fh | PMADRH | — | — | — | — | — | PMADRH<1:0> | | | ---- --00 | ---- --00 | | |
| 190h | PMDATL | Program Memory Data Register Low Byte | | | | | | | | | 0000 0000 | 0000 0000 | |
| 191h | PMDATH | — | — | Program Memory Data Register High Byte | | | | | | | | --00 0000 | --00 0000 |
| 192h | COG1PH | — | — | — | — | G1PH<3:0> | | | | ---- xxxx | ---- uuuu | | |
| 193h | COG1BLK | G1BLKR<3:0> | | | | G1BLKF<3:0> | | | | xxxx xxxx | uuuu uuuu | | |
| 194h | COG1DB | G1DBR<3:0> | | | | G1DBF<3:0> | | | | xxxx xxxx | uuuu uuuu | | |
| 195h | COG1CON0 | G1EN | G1OE1 | G1OE0 | G1POL1 | G1POL0 | G1LD | G1CS<1:0> | | | 0000 0000 | 0000 0000 | |
| 196h | COG1CON1 | G1FSIM | G1RSIM | G1FS<2:0> | | | G1RS<2:0> | | | 0000 0000 | 0000 0000 | | |
| 197h | COG1ASD | G1ASDE | G1ARSEN | G1ASDL1 | G1ASDL0 | G1ASDSHLT | G1ASDSC2 | G1ASDSC1 | G1ASDSFLT | 0000 0000 | 0000 0000 | | |
| 198h to 19Fh | — | Unimplemented | | | | | | | | | — | — | |

Legend: — = Unimplemented locations read as '0', u = unchanged, x = unknown, q = value depends on condition shaded = unimplemented

Note 1: Other (non Power-up) Resets include MCLR Reset and Watchdog Timer Reset during normal operation.

Note 2: MCLR and WDT Reset does not affect the previous value data latch. The IOCIF bit will be cleared upon Reset but will set again if the mismatch exists.

PIC12F752/HV752

2.2.2.1 STATUS Register

The STATUS register, shown in [Register 2-1](#), contains:

- the arithmetic status of the ALU
- the Reset status
- the bank select bits for data memory (RAM)

The STATUS register can be the destination for any instruction, like 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 TO and 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`, `SWAPF` and `MOVWF` instructions are used to alter the STATUS register, because these instructions do not affect any Status bits. For other instructions not affecting any Status bits, see the [Section 16.0 "Instruction Set Summary"](#).

REGISTER 2-1: STATUS: STATUS REGISTER

| | | | | | | | |
|-------|-------|-------|------------------------|------------------------|-------|-------------------|------------------|
| R/W-0 | R/W-0 | R/W-0 | R-1 | R-1 | R/W-x | R/W-x | R/W-x |
| IRP | RP1 | RP0 | $\overline{\text{TO}}$ | $\overline{\text{PD}}$ | Z | DC ⁽¹⁾ | C ⁽¹⁾ |
| bit 7 | | | | | | | bit 0 |

Legend:

| | | |
|-------------------|------------------|------------------------------------|
| R = Readable bit | W = Writable bit | U = Unimplemented bit, read as '0' |
| -n = Value at POR | '1' = Bit is set | '0' = Bit is cleared |
| | | x = Bit is unknown |

- bit 7 **IRP:** Register Bank Select bit (used for indirect addressing)
 1 = Bank 2, 3 (100h-1FFh)
 0 = Bank 0, 1 (00h-FFh)
- bit 6 **RP1:** Register Bank Select bit (used for direct addressing)
 00 = Bank 0 (00h-7Fh)
 01 = Bank 1 (80h-FFh)
 10 = Bank 2 (100h-17Fh)
 11 = Bank 3 (180h-1FFh)
- bit 5 **RP0:** Register Bank Select bit (used for direct addressing)
 1 = Bank 1 (80h-FFh)
 0 = Bank 0 (00h-7Fh)
- bit 4 **TO:** Time-Out bit
 1 = After power-up, `CLRWDT` instruction or `SLEEP` instruction
 0 = A WDT time-out occurred
- bit 3 **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/Borrow bit⁽²⁾ (`ADDWF`, `ADDLW`, `SUBLW`, `SUBWF` instructions), For Borrow, the polarity is reversed.
 1 = A carry-out from the 4th low-order bit of the result occurred
 0 = No carry-out from the 4th low-order bit of the result
- bit 0 **C:** Carry/Borrow bit⁽²⁾ (`ADDWF`, `ADDLW`, `SUBLW`, `SUBWF` instructions)
 1 = A carry-out from the Most Significant bit of the result occurred
 0 = No carry-out from the Most Significant bit of the result occurred

- Note 1:** The C and DC bits operate as a Borrow and Digit Borrow out bit, respectively, in subtraction. See the `SUBLW` and `SUBWF` instructions for examples.
- Note 2:** For Borrow, the polarity is reversed. A subtraction is executed by adding the two's complement of the second operand. For rotate (`RRF`, `RLF`) instructions, this bit is loaded with either the high-order or low-order bit of the source register.

2.2.2.2 OPTION Register

The OPTION register is a readable and writable register, which contains various control bits to configure:

- Timer0/WDT prescaler
- External RA2/INT interrupt
- Timer0
- Weak pull-ups on PORTA

Note: To achieve a 1:1 prescaler assignment for Timer0, assign the prescaler to the WDT by setting PSA bit to '1' of the OPTION register. See **Section 6.1.3 "Software Programmable Prescaler"**.

REGISTER 2-2: OPTION_REG: OPTION REGISTER

| | | | | | | | |
|--------------------------|--------|-------|-------|-------|---------|-------|-------|
| R/W-1 | R/W-1 | R/W-1 | R/W-1 | R/W-1 | R/W-1 | R/W-1 | R/W-1 |
| $\overline{\text{RAPU}}$ | INTEDG | T0CS | T0SE | PSA | PS<2:0> | | |
| bit 7 | | | | | | bit 0 | |

Legend:

| | | |
|-------------------|------------------|--|
| R = Readable bit | W = Writable bit | U = Unimplemented bit, read as '0' |
| -n = Value at POR | '1' = Bit is set | '0' = Bit is cleared x = Bit is unknown |

- bit 7 **$\overline{\text{RAPU}}$:** PORTA Pull-up Enable bit
 1 = PORTA pull-ups are disabled
 0 = PORTA pull-ups are enabled by individual PORT latch values
- bit 6 **INTEDG:** Interrupt Edge Select bit
 1 = Interrupt on rising edge of INT pin
 0 = Interrupt on falling edge of INT pin
- bit 5 **T0CS:** Timer0 Clock Source Select bit
 1 = Transition on T0CKI pin
 0 = Internal instruction cycle clock (FOSC/4)
- bit 4 **T0SE:** Timer0 Source Edge Select bit
 1 = Increment on high-to-low transition on T0CKI pin
 0 = Increment on low-to-high transition on T0CKI pin
- bit 3 **PSA:** Prescaler Assignment bit
 1 = Prescaler is assigned to the WDT
 0 = Prescaler is assigned to the Timer0 module
- bit 2-0 **PS<2:0>:** 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 |

PIC12F752/HV752

2.2.2.3 INTCON Register

The INTCON register is a readable and writable register, which contains the various enable and flag bits for TMR0 register overflow, IOCIE change and external RA2/INT pin interrupts.

Note: Interrupt flag bits are set when an interrupt condition occurs, regardless of the state of its corresponding enable bit or the Global Enable bit, GIE of the INTCON register. User software should ensure the appropriate interrupt flag bits are clear prior to enabling an interrupt.

REGISTER 2-3: INTCON: INTERRUPT CONTROL REGISTER

| R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 |
|-------|-------|-------|-------|-------|-------|-------|-------|
| GIE | PEIE | T0IE | INTE | IOCIE | T0IF | INTF | IOCIF |
| bit 7 | | | | | | | bit 0 |

Legend:

R = Readable bit

W = Writable bit

U = Unimplemented bit, read as '0'

-n = Value at POR

'1' = Bit is set

'0' = Bit is cleared

x = Bit is unknown

- bit 7 **GIE:** Global Interrupt Enable bit
1 = Enables all unmasked interrupts
0 = Disables all interrupts
- bit 6 **PEIE:** Peripheral Interrupt Enable bit
1 = Enables all unmasked peripheral interrupts
0 = Disables all peripheral interrupts
- bit 5 **T0IE:** Timer0 Overflow Interrupt Enable bit
1 = Enables the Timer0 interrupt
0 = Disables the Timer0 interrupt
- bit 4 **INTE:** RA2/INT External Interrupt Enable bit
1 = Enables the RA2/INT external interrupt
0 = Disables the RA2/INT external interrupt
- bit 3 **IOCIE:** Interrupt-on-Change Interrupt Enable bit⁽¹⁾
1 = Enables the IOC change interrupt
0 = Disables the IOC change interrupt
- bit 2 **T0IF:** Timer0 Overflow Interrupt Flag bit⁽²⁾
1 = Timer0 register has overflowed (must be cleared in software)
0 = Timer0 register did not overflow
- bit 1 **INTF:** RA2/INT External Interrupt Flag bit
1 = The RA2/INT external interrupt occurred (must be cleared in software)
0 = The RA2/INT external interrupt did not occur
- bit 0 **IOCIF:** Interrupt-on-Change Interrupt Flag bit
1 = An IOC pin has changed state and generated an interrupt
0 = No pin interrupts have been generated

Note 1: IOC register must also be enabled.

Note 2: T0IF bit is set when TMR0 rolls over. TMR0 is unchanged on Reset and should be initialized before clearing T0IF bit.

2.2.2.4 PIE1 Register

The PIE1 register contains the Peripheral Interrupt Enable bits, as shown in [Register 2-4](#).

Note: Bit PEIE of the INTCON register must be set to enable any peripheral interrupt.

REGISTER 2-4: PIE1: PERIPHERAL INTERRUPT ENABLE REGISTER 1

| | | | | | | | |
|---------|-------|-----|-----|-----|----------|--------|--------|
| R/W-0 | R/W-0 | U-0 | U-0 | U-0 | R/W-0 | R/W-0 | R/W-0 |
| TMR1GIE | ADIE | — | — | — | HLTMR1IE | TMR2IE | TMR1IE |
| bit 7 | | | | | | | bit 0 |

Legend:

| | | |
|-------------------|------------------|------------------------------------|
| R = Readable bit | W = Writable bit | U = Unimplemented bit, read as '0' |
| -n = Value at POR | '1' = Bit is set | '0' = Bit is cleared |
| | | x = Bit is unknown |

- bit 7 **TMR1GIE:** ADC Interrupt Enable bit
 1 = Enables the TMR1 gate interrupt
 0 = Disables the TMR1 gate interrupt

- bit 6 **ADIE:** ADC Interrupt Enable bit
 1 = Enables the ADC interrupt
 0 = Disables the ADC interrupt

- bit 5-3 **Unimplemented:** Read as '0'

- bit 2 **HLTMR1IE:** Hardware Limit Timer1 Interrupt Enable bit
 1 = Enables the HLTMR1 interrupt
 0 = Disables the HLTMR1 interrupt

- bit 1 **TMR2IE:** Timer2 Interrupt Enable bit
 1 = Enables the Timer2 interrupt
 0 = Disables the Timer2 interrupt

- bit 0 **TMR1IE:** Timer1 Interrupt Enable bit
 1 = Enables the Timer1 interrupt
 0 = Disables the Timer1 interrupt

PIC12F752/HV752

2.2.2.5 PIE2 Register

The PIE2 register contains the Peripheral Interrupt Enable bits, as shown in [Register 2-5](#).

Note: Bit PEIE of the INTCON register must be set to enable any peripheral interrupt.

REGISTER 2-5: PIE2: PERIPHERAL INTERRUPT ENABLE REGISTER 1

| | | | | | | | |
|-------|-----|-------|-------|-----|--------|-------|--------|
| U-0 | U-0 | R/W-0 | R/W-0 | U-0 | R/W-0 | U-0 | R/W-0 |
| — | — | C2IE | C1IE | — | COG1IE | — | CCP1IE |
| bit 7 | | | | | | bit 0 | |

Legend:

| | | |
|-------------------|------------------|------------------------------------|
| R = Readable bit | W = Writable bit | U = Unimplemented bit, read as '0' |
| -n = Value at POR | '1' = Bit is set | '0' = Bit is cleared |
| | | x = Bit is unknown |

- bit 7-6 **Unimplemented:** Read as '0'
- bit 5 **C2IE:** Comparator 2 Interrupt Enable bit
 - 1 = Enables the Comparator 2 interrupt
 - 0 = Disables the Comparator 2 interrupt
- bit 4 **C1IE:** Comparator 1 Interrupt Enable bit
 - 1 = Enables the Comparator 1 interrupt
 - 0 = Disables the Comparator 1 interrupt
- bit 3 **Unimplemented:** Read as '0'
- bit 2 **COG1IE:** COG 1 Interrupt Flag bit
 - 1 = COG1 interrupt enabled
 - 0 = COG1 interrupt disabled
- bit 1 **Unimplemented:** Read as '0'
- bit 0 **CCP1IE:** CCP1 Interrupt Enable bit
 - 1 = Enables the CCP1 interrupt
 - 0 = Disables the CCP1 interrupt

2.2.2.6 PIR1 Register

The PIR1 register contains the Peripheral Interrupt flag bits, as shown in [Register 2-6](#).

Note: Interrupt flag bits are set when an interrupt condition occurs, regardless of the state of its corresponding enable bit or the Global Enable bit, GIE of the INTCON register. User software should ensure the appropriate interrupt flag bits are clear prior to enabling an interrupt.

REGISTER 2-6: PIR1: PERIPHERAL INTERRUPT REQUEST REGISTER 1

| | | | | | | | |
|---------|-------|-----|-----|-----|----------|--------|--------|
| R/W-0 | R/W-0 | U-0 | U-0 | U-0 | R/W-0 | R/W-0 | R/W-0 |
| TMR1GIF | ADIF | — | — | — | HLTMR1IF | TMR2IF | TMR1IF |
| bit 7 | | | | | | | bit 0 |

Legend:

| | | |
|-------------------|------------------|--|
| R = Readable bit | W = Writable bit | U = Unimplemented bit, read as '0' |
| -n = Value at POR | '1' = Bit is set | '0' = Bit is cleared x = Bit is unknown |

- bit 7 **TMR1GIF:** TMR1 Gate Interrupt Flag bit
 1 = Timer1 gate interrupt is pending
 0 = Timer1 gate interrupt is not pending
- bit 6 **ADIF:** ADC Interrupt Flag bit
 1 = ADC conversion complete
 0 = ADC conversion has not completed or has not been started
- bit 5-3 **Unimplemented:** Read as '0'
- bit 2 **HLTMR1IF:** Hardware Limit Timer1 to HLTMR1 Match Interrupt Flag bit
 1 = HLTMR1 to HLTMR1 match occurred (must be cleared in software)
 0 = HLTMR1 to HLTMR1 match did not occur
- bit 1 **TMR2IF:** Timer2 to PR2 Match Interrupt Flag bit
 1 = Timer2 to PR2 match occurred (must be cleared in software)
 0 = Timer2 to PR2 match did not occur
- bit 0 **TMR1IF:** Timer1 Interrupt Flag bit
 1 = Timer1 rolled over (must be cleared in software)
 0 = Timer1 has not rolled over

PIC12F752/HV752

2.2.2.7 PIR2 Register

The PIR2 register contains the Peripheral Interrupt flag bits, as shown in [Register 2-7](#).

Note: Interrupt flag bits are set when an interrupt condition occurs, regardless of the state of its corresponding enable bit or the Global Enable bit, GIE of the INTCON register. User software should ensure the appropriate interrupt flag bits are clear prior to enabling an interrupt.

REGISTER 2-7: PIR2: PERIPHERAL INTERRUPT REQUEST REGISTER 1

| | | | | | | | |
|-------|-----|-------|-------|-----|--------|-----|--------|
| U-0 | U-0 | R/W-0 | R/W-0 | U-0 | R/W-0 | U-0 | R/W-0 |
| — | — | C2IF | C1IF | | COG1IF | — | CCP1IF |
| bit 7 | | | | | | | bit 0 |

Legend:

| | | |
|-------------------|------------------|--|
| R = Readable bit | W = Writable bit | U = Unimplemented bit, read as '0' |
| -n = Value at POR | '1' = Bit is set | '0' = Bit is cleared x = Bit is unknown |

- bit 7-6 **Unimplemented:** Read as '0'
- bit 5 **C2IF:** Comparator 1 Interrupt Flag bit
 - 1 = Comparator output (C2OUT bit) has changed (must be cleared in software)
 - 0 = Comparator output (C2OUT bit) has not changed
- bit 4 **C1IF:** Comparator 1 Interrupt Flag bit
 - 1 = Comparator output (C1OUT bit) has changed (must be cleared in software)
 - 0 = Comparator output (C1OUT bit) has not changed
- bit 3 **Unimplemented:** Read as '0'
- bit 2 **COG1IF:** COG 1 Interrupt Flag bit
 - 1 = COG1 has generated an auto-shutdown interrupt
 - 0 = COG1 has NOT generated an auto-shutdown interrupt
- bit 1 **Unimplemented:** Read as '0'
- bit 0 **CCP1IF:** ECCP Interrupt Flag bit
 - Capture Mode
 - 1 = A TMR1 register capture occurred (must be cleared in software)
 - 0 = No TMR1 register capture occurred
 - Compare Mode
 - 1 = A TMR1 register compare match occurred (must be cleared in software)
 - 0 = No TMR1 register compare match occurred
 - PWM mode
Unused in this mode

2.2.2.8 PCON Register

The Power Control (PCON) register (see [Table 17-2](#)) contains flag bits to differentiate between a:

- Power-on Reset ($\overline{\text{POR}}$)
- Brown-out Reset ($\overline{\text{BOR}}$)
- Watchdog Timer Reset (WDT)
- External MCLR Reset

The PCON register also controls the software enable of the $\overline{\text{BOR}}$.

The PCON register bits are shown in [Register 2-8](#).

REGISTER 2-8: PCON: POWER CONTROL REGISTER

| | | | | | | | |
|-------|-----|-----|-----|-----|-----|-------------------------|-------------------------|
| U-0 | U-0 | U-0 | U-0 | U-0 | U-0 | R/W-q/u | R/W-q/u |
| — | — | — | — | — | — | $\overline{\text{POR}}$ | $\overline{\text{BOR}}$ |
| bit 7 | | | | | | bit 0 | |

Legend:

R = Readable bit

W = Writable bit

U = Unimplemented bit, read as '0'

u = Bit is unchanged

x = Bit is unknown

-n/n = Value at POR and BOR/Value at all other Resets

'1' = Bit is set

'0' = Bit is cleared

q = unchanged

bit 7-2

Unimplemented: Read as '0'

bit 1

$\overline{\text{POR}}$: Power-on Reset Status bit

1 = No Power-on Reset occurred

0 = A Power-on Reset occurred (must be set in software after a Power-on Reset occurs)

bit 0

$\overline{\text{BOR}}$: Brown-out Reset Status bit

1 = No Brown-out Reset occurred

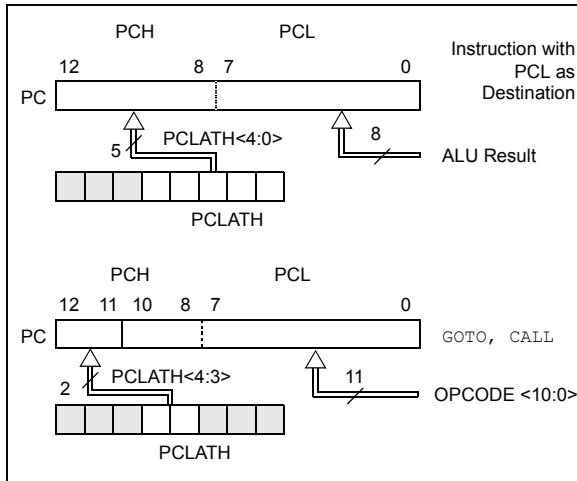
0 = A Brown-out Reset occurred (must be set in software after a Brown-out Reset occurs)

PIC12F752/HV752

2.3 PCL and PCLATH

The Program Counter (PC) is 13 bits wide. The low byte comes from the PCL register, which is a readable and writable register. The high byte (PC<12:8>) is not directly readable or writable and comes from PCLATH. On any Reset, the PC is cleared. Figure 2-3 shows the two situations for the loading of the PC. The upper example in Figure 2-3 shows how the PC is loaded on a write to PCL (PCLATH<4:0> → PCH). The lower example in Figure 2-3 shows how the PC is loaded during a CALL or GOTO instruction (PCLATH<4:3> → PCH).

FIGURE 2-3: LOADING OF PC IN DIFFERENT SITUATIONS



2.3.1 MODIFYING PCL

Executing any instruction with the PCL register as the destination simultaneously causes the Program Counter PC<12:8> bits (PCH) to be replaced by the contents of the PCLATH register. This allows the entire contents of the program counter to be changed by writing the desired upper 5 bits to the PCLATH register. When the lower 8 bits are written to the PCL register, all 13 bits of the program counter will change to the values contained in the PCLATH register and those being written to the PCL register.

A computed GOTO is accomplished by adding an offset to the program counter (ADDWF PCL). Care should be exercised when jumping into a look-up table or program branch table (computed GOTO) by modifying the PCL register. Assuming that PCLATH is set to the table start address, if the table length is greater than 255 instructions or if the lower 8 bits of the memory address rolls over from 0xFF to 0x00 in the middle of the table, then PCLATH must be incremented for each address rollover that occurs between the table beginning and the target location within the table.

For more information refer to Application Note AN556, "Implementing a Table Read" (DS00556).

2.3.2 STACK

The PIC12F752/HV752 Family has an 8-level x 13-bit wide hardware stack (see Figure 2-1). The stack space is not part of either program or data space and the Stack Pointer is not readable or writable. The PC is PUSHed onto the stack when a CALL instruction is executed or an interrupt causes a branch. The stack is POPed in the event of a RETURN, RETLW or RETFIE instruction execution. PCLATH is not affected by a PUSH or POP operation.

The stack operates as a circular buffer. This means that after the stack has been PUSHed eight times, the ninth push overwrites the value that was stored from the first push. The tenth push overwrites the second push (and so on).

- Note 1:** There are no Status bits to indicate Stack Overflow 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, RETURN, RETLW and RETFIE instructions or the vectoring to an interrupt address.

2.4 Indirect Addressing, INDF and FSR Registers

The INDF register is not a physical register. Addressing the INDF register will cause indirect addressing.

Indirect addressing is possible by using the INDF register. Any instruction using the INDF register actually accesses data pointed to by the File Select Register (FSR). Reading INDF itself indirectly will produce 00h. Writing to the INDF register indirectly results in a no operation (although Status bits may be affected). An effective 9-bit address is obtained by concatenating the 8-bit FSR and the IRP bit of the STATUS register, as shown in Figure 2-4.

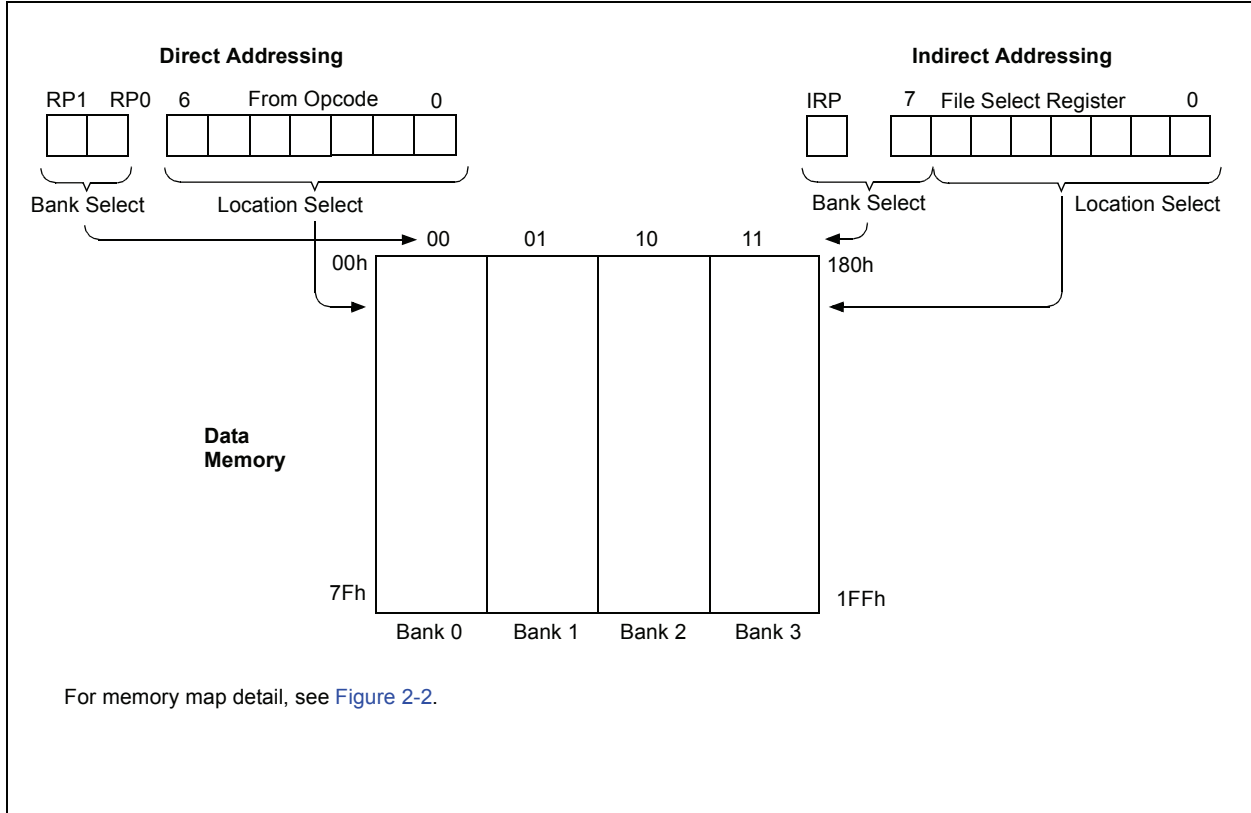
A simple program to clear RAM location 40h-7Fh using indirect addressing is shown in Example 2-1.

EXAMPLE 2-1: INDIRECT ADDRESSING

```

MOV LW 0x40 ;initialize pointer
MOV WF FSR ;to RAM
NEXT CLRF INDF ;clear INDF register
INCF FSR ;inc pointer
BTFSS FSR,7 ;all done?
GOTO NEXT ;no clear next
CONTINUE ;yes continue
    
```

FIGURE 2-4: DIRECT/INDIRECT ADDRESSING PIC12F752/HV752



PIC12F752/HV752

NOTES:

3.0 FLASH PROGRAM MEMORY SELF READ/SELF WRITE CONTROL

The Flash program memory is readable and writable during normal operation (full VDD range). This memory is not directly mapped in the register file space. Instead, it is indirectly addressed through the Special Function Registers (see Registers 3-1 to 3-5). There are six SFRs used to read and write this memory:

- PMCON1
- PMCON2
- PMDATL
- PMDATH
- PMADRL
- PMADRH

When interfacing the program memory block, the PMDATL and PMDATH registers form a two-byte word which holds the 14-bit data for read/write, and the PMADRL and PMADRH registers form a two-byte word which holds the 10-bit address of the Flash location being accessed. These devices have 1K words of program Flash with an address range from 0000h to 03FFh.

The program memory allows single word read and a by four word write. A four word write automatically erases the row of the location and writes the new data (erase before write).

The write time is controlled by an on-chip timer. The write/erase voltages are generated by an on-chip charge pump rated to operate over the voltage range of the device for byte or word operations.

When the device is code-protected, the CPU may continue to read and write the Flash program memory.

Depending on the settings of the Flash Program Memory Enable (WRT<1:0>) bits, the device may or may not be able to write certain blocks of the program memory, however, reads of the program memory are allowed.

When the Flash program memory Code Protection (CP) bit in the Configuration Word register is enabled, the program memory is code-protected, and the device programmer (ICSP™) cannot access data or program memory.

3.1 PMADRH and PMADRL Registers

The PMADRH and PMADRL registers can address up to a maximum of 1K words of program memory.

When selecting a program address value, the Most Significant Byte (MSB) of the address is written to the PMADRH register and the Least Significant Byte (LSB) is written to the PMADRL register.

3.2 PMCON1 and PMCON2 Registers

PMCON1 is the control register for the data program memory accesses.

Control bits RD and WR initiate read and write, respectively. These bits cannot be cleared, only set in software. They are cleared in hardware at completion of the read or write operation. The inability to clear the WR bit in software prevents the accidental premature termination of a write operation.

The WREN bit, when set, will allow a write operation. On power-up, the WREN bit is clear.

PMCON2 is not a physical register. Reading PMCON2 will read all '0's. The PMCON2 register is used exclusively in the Flash memory write sequence.

PIC12F752/HV752

3.3 Flash Program Memory Control Registers

REGISTER 3-1: PMDATL: PROGRAM MEMORY DATA LOW BYTE REGISTER

| | | | | | | | |
|-------------|-------|-------|-------|-------|-------|-------|-------|
| R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 |
| PMDATL<7:0> | | | | | | | |
| bit 7 | | | | | | | bit 0 |

Legend:

R = Readable bit W = Writable bit U = Unimplemented bit, read as '0'
 -n = Value at POR '1' = Bit is set '0' = Bit is cleared x = Bit is unknown

bit 7-0 **PMDATL<7:0>**: 8 Least Significant Data bits to Write or Read from Program Memory

REGISTER 3-2: PMADRL: PROGRAM MEMORY ADDRESS LOW BYTE REGISTER

| | | | | | | | |
|-------------|-------|-------|-------|-------|-------|-------|-------|
| R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 |
| PMADRL<7:0> | | | | | | | |
| bit 7 | | | | | | | bit 0 |

Legend:

R = Readable bit W = Writable bit U = Unimplemented bit, read as '0'
 -n = Value at POR '1' = Bit is set '0' = Bit is cleared x = Bit is unknown

bit 7-0 **PMADRL<7:0>**: 8 Least Significant Address bits for Program Memory Read/Write Operation

REGISTER 3-3: PMDATH: PROGRAM MEMORY DATA HIGH BYTE REGISTER

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

Legend:

R = Readable bit W = Writable bit U = Unimplemented bit, read as '0'
 -n = Value at POR '1' = Bit is set '0' = Bit is cleared x = Bit is unknown

bit 7-6 **Unimplemented:** Read as '0'

bit 5-0 **PMDATH<5:0>**: 6 Most Significant Data bits from Program Memory

REGISTER 3-4: PMADRH: PROGRAM MEMORY ADDRESS HIGH BYTE REGISTER

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

Legend:

R = Readable bit W = Writable bit U = Unimplemented bit, read as '0'
 -n = Value at POR '1' = Bit is set '0' = Bit is cleared x = Bit is unknown

bit 7-2 **Unimplemented:** Read as '0'

bit 1-0 **PMADRH<1:0>**: Specifies the 2 Most Significant Address bits or High bits for Program Memory Reads.

REGISTER 3-5: PMCON1: PROGRAM MEMORY CONTROL 1 REGISTER

| | | | | | | | |
|-------|-----|-----|-----|-----|---------|------------|------------|
| U-0 | U-0 | U-0 | U-0 | U-0 | R/W-0/0 | R/S/HC-0/0 | R/S/HC-0/0 |
| — | — | — | — | — | WREN | WR | RD |
| bit 7 | | | | | | bit 0 | |

Legend:

| | | |
|-------------------------|----------------------|---|
| R = Readable bit | W = Writable bit | U = Unimplemented bit, read as '0' |
| S = Bit can only be set | x = Bit is unknown | -n/n = Value at POR and BOR/Value at all other Resets |
| '1' = Bit is set | '0' = Bit is cleared | HC = Bit is cleared by hardware |

bit 7-3 **Unimplemented:** Read as '0'

bit 2 **WREN:** Program/Erase Enable bit
 1 = Allows program/erase cycles
 0 = Inhibits programming/erasing of program Flash

bit 1 **WR:** Write Control bit
 1 = Initiates a program Flash program/erase operation
 The operation is self-timed and the bit is cleared by hardware once operation is complete.
 The WR bit can only be set (not cleared) in software.
 0 = Program/erase operation to the Flash is complete and inactive

bit 0 **RD:** Read Control bit
 1 = Initiates a program Flash read. Read takes one cycle. RD is cleared in hardware. The RD bit can only be set (not cleared) in software.
 0 = Does not initiate a program Flash read

Note 1: Unimplemented bit, read as '1'.

PIC12F752/HV752

3.4 Reading the Flash Program Memory

To read a program memory location, the user must write two bytes of the address to the PMADRL and PMADRH registers, and then set control bit RD (PMCON1<0>). Once the read control bit is set, the program memory Flash controller will use the second instruction cycle after to read the data. This causes the second instruction immediately following the “BSF PMCON1, RD” instruction to be ignored. The data is available in the very next cycle in the PMDATL and PMDATH registers; it can be read as two bytes in the following instructions. PMDATL and PMDATH registers will hold this value until another read or until it is written to by the user (during a write operation).

EXAMPLE 3-1: FLASH PROGRAM READ

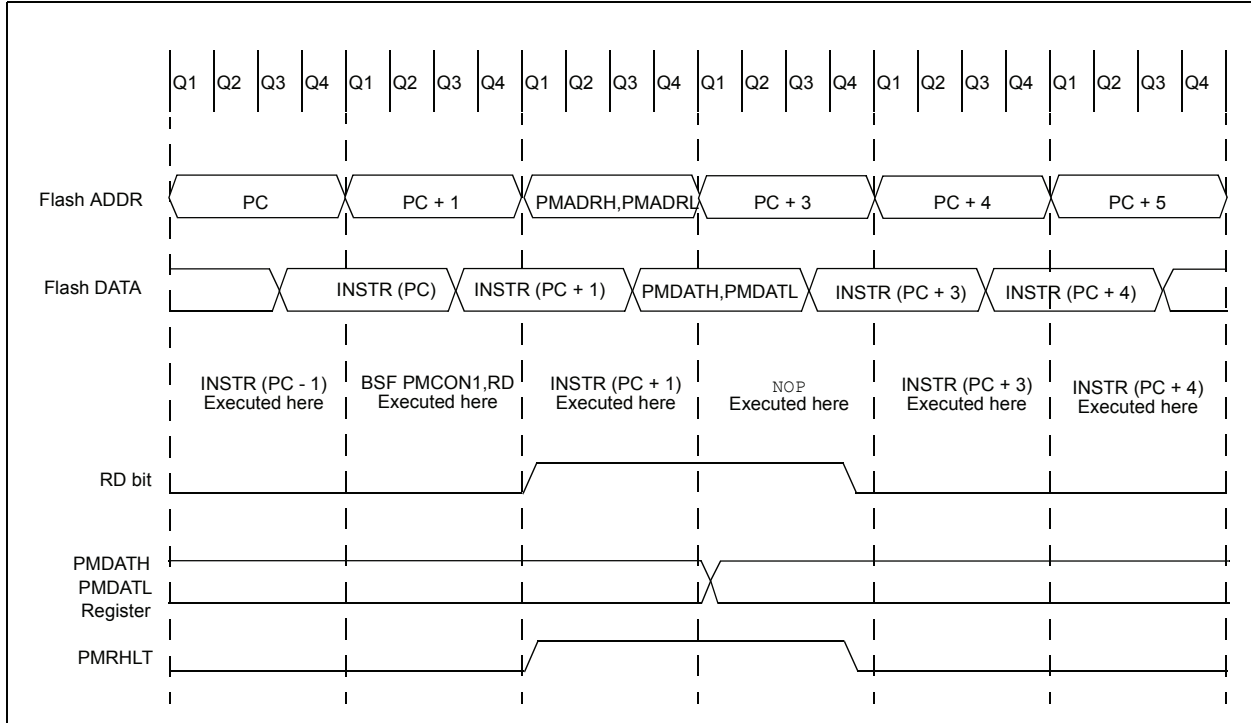
```
BANKSEL  PM_ADR           ; Change STATUS bits RP1:0 to select bank with PMADRL
MOVLW    MS_PROG_PM_ADDR ;
MOVWF    PMADRH          ; MS Byte of Program Address to read
MOVLW    LS_PROG_PM_ADDR ;
MOVWF    PMADRL          ; LS Byte of Program Address to read
BANKSEL  PMCON1          ; Bank to containing PMCON1
BSF      PMCON1, RD      ; PM Read

NOP                                             ; First instruction after BSF PMCON1,RD executes normally

NOP                                             ; Any instructions here are ignored as program
                                             ; memory is read in second cycle after BSF PMCON1,RD
                                             ;

BANKSEL  PMDATL          ; Bank to containing PMADRL
MOVF     PMDATL, W       ; W = LS Byte of Program PMDATL
MOVF     PMDATH, W       ; W = MS Byte of Program PMDATL
```


FIGURE 3-1: FLASH PROGRAM MEMORY READ CYCLE EXECUTION



3.5 Writing the Flash Program Memory

A word of the Flash program memory may only be written to if the word is in an unprotected segment of memory.

Flash program memory must be written in four-word blocks. See [Figure 3-2](#) and [Figure 3-3](#) for more details. A block consists of four words with sequential addresses, with a lower boundary defined by an address, where $PMADRL<1:0> = 00$. All block writes to program memory are done as 16-word erase by four-word write operations. The write operation is edge-aligned and cannot occur across boundaries.

To write program data, it must first be loaded into the buffer registers (see [Figure 3-2](#)). This is accomplished by first writing the destination address to PMADRL and PMADRH and then writing the data to PMDATL and PMDATH. After the address and data have been set up, then the following sequence of events must be executed:

1. Write 55h, then AAh, to PMCON2 (Flash programming sequence).
2. Set the WR control bit of the PMCON1 register.

All four buffer register locations should be written to with correct data. If less than four words are being written to in the block of four words, then a read from the program memory location(s) not being written to must be performed. This takes the data from the program location(s) not being written and loads it into the PMDATL and PMDATH registers. Then the sequence of events to transfer data to the buffer registers must be executed.

To transfer data from the buffer registers to the program memory, the PMADRL and PMADRH must point to the last location in the four-word block ($PMADRL<1:0> = 11$). Then the following sequence of events must be executed:

1. Write 55h, then AAh, to PMCON2 (Flash programming sequence).
2. Set control bit WR of the PMCON1 register to begin the write operation.

The user must follow the same specific sequence to initiate the write for each word in the program block, writing each program word in sequence (000, 001, 010, 011). When the write is performed on the last word ($PMADRL<1:0> = 11$), a block of sixteen words is automatically erased and the content of the four-word buffer registers are written into the program memory.

After the “BSF PMCON1, WR” instruction, the processor requires two cycles to set up the erase/write operation. The user must place two NOP instructions after the WR bit is set. Since data is being written to buffer registers, the writing of the first three words of the block appears to occur immediately. The processor will halt internal operations for the typical 4 ms, only during the cycle in

which the erase takes place (i.e., the last word of the sixteen-word block erase). This is not Sleep mode as the clocks and peripherals will continue to run. After the four-word write cycle, the processor will resume operation with the third instruction after the PMCON1 write instruction. The above sequence must be repeated for the higher 12 words.

3.6 Protection Against Spurious Write

There are conditions when the device should not write to the program memory. To protect against spurious writes, various mechanisms have been built in. On power-up, WREN is cleared. Also, the Power-up Timer (64 ms duration) prevents program memory writes.

The write initiate sequence and the WREN bit help prevent an accidental write during brown-out, power glitch or software malfunction.

3.7 Operation During Code-Protect

When the device is code-protected, the CPU is able to read and write unscrambled data to the program memory.

3.8 Operation During Write Protect

When the program memory is write-protected, the CPU can read and execute from the program memory. The portions of program memory that are write protected can be modified by the CPU using the PMCON registers, but the protected program memory cannot be modified using ICSP mode.

FIGURE 3-2: BLOCK WRITES TO 1K FLASH PROGRAM MEMORY

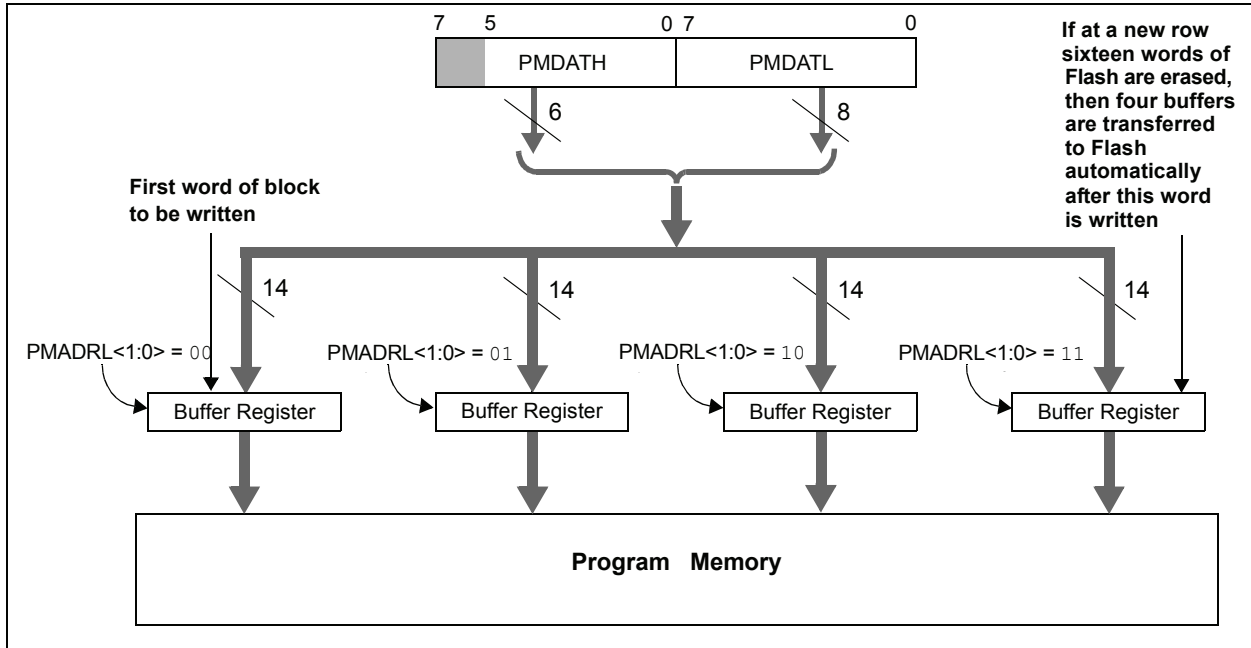
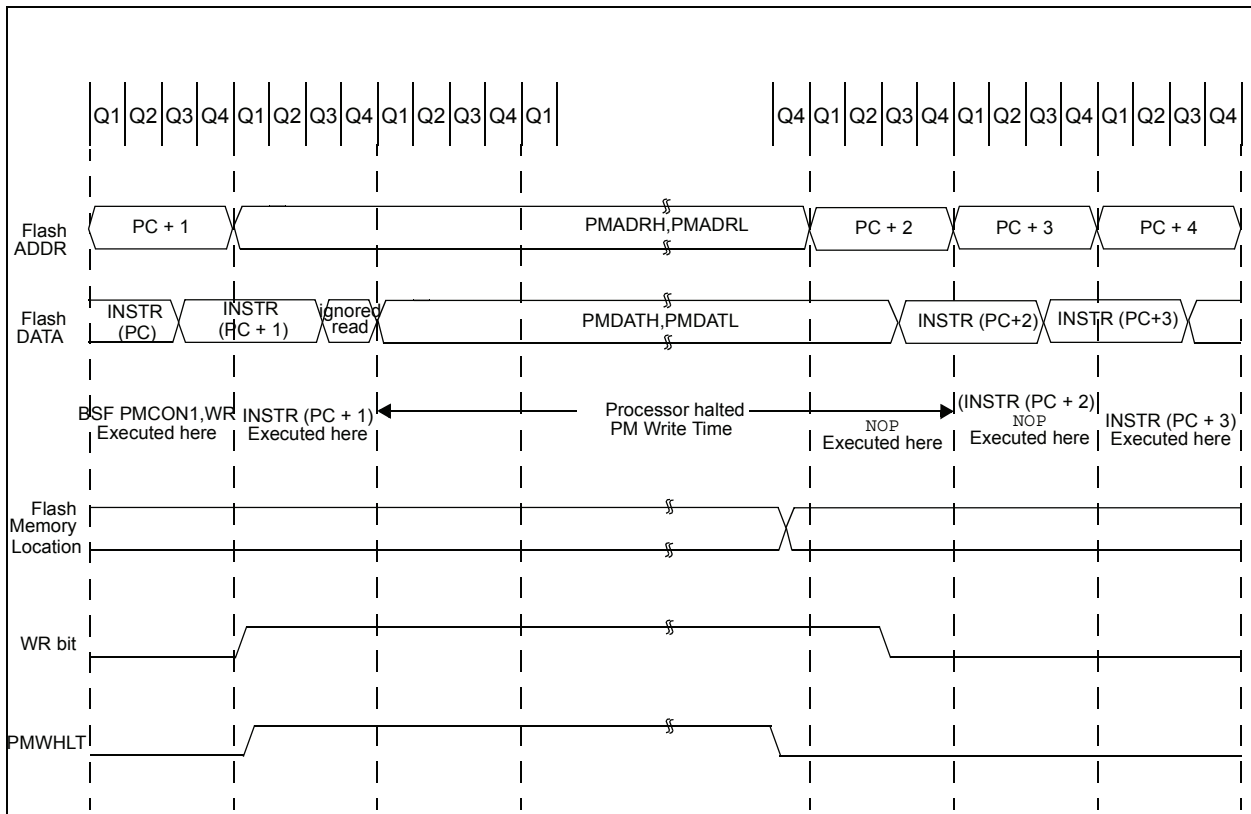


FIGURE 3-3: FLASH PROGRAM MEMORY LONG WRITE CYCLE EXECUTION



PIC12F752/HV752

An example of the complete four-word write sequence is shown in [Example 3-2](#). The initial address is loaded into the PMADRH and PMADRL register pair; the four words of data are loaded using indirect addressing.

EXAMPLE 3-2: WRITING TO FLASH PROGRAM MEMORY

```
;/;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;/
; This write routine assumes the following:
;   A valid starting address (the least significant bits = '00')
;   is loaded in ADDRH:ADDRL
;   ADDRH, ADDRL and DATADDR are all located in data memory
;
BANKSEL          PMADRH
MOVF  ADDRH,W    ;Load initial address
MOVWF PMADRH    ;
MOVF  ADDRL,W   ;
MOVWF PMADRL    ;
MOVF  DATAADDR,W ;Load initial data address
MOVWF FSR       ;
LOOP MOVF INDF,W ;Load first data byte into lower
MOVWF PMDATL    ;
INCF  FSR,F     ;Next byte
MOVF  INDF,W   ;Load second data byte into upper
MOVWF PMDATH    ;
INCF  FSR,F     ;
BANKSEL PMCON1
BSF  PMCON1,WREN ;Enable writes
BCF  INTCON,GIE ;Disable interrupts (if using)
BTFSC INTCON,GIE ;See AN576
GOTO $-2
;/;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;/
; Required Sequence
MOVLW 55h      ;Start of required write sequence:
MOVWF PMCON2   ;Write 55h
MOVLW 0AAh    ;
MOVWF PMCON2   ;Write 0AAh
BSF  PMCON1,WR ;Set WR bit to begin write
NOP   ;Required to transfer data to the buffer
NOP   ;registers
;/;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;/
BCF  PMCON1,WREN ;Disable writes
BSF  INTCON,GIE  ;Enable interrupts (comment out if not using interrupts)
BANKSEL PMADRL
MOVF  PMADRL, W
INCF  PMADRL,F  ;Increment address
ANDLW 0x03     ;Indicates when sixteen words have been programmed
SUBLW 0x03     ;Change value for different size write blocks
;0x0F = 16 words
;0x0B = 12 words
;0x07 = 8 words
;0x03 = 4 words
BTFSS STATUS,Z ;Exit on a match,
GOTO  LOOP     ;Continue if more data needs to be written
```

TABLE 3-1: SUMMARY OF REGISTERS ASSOCIATED WITH FLASH PROGRAM MEMORY

| Name | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | Register on Page |
|--------|-----------------------------------|-------|-------------|-------|-------|-------|-------------|-------|------------------|
| PMCON1 | — | — | — | — | — | WREN | WR | RD | 31 |
| PMCON2 | Program Memory Control Register 2 | | | | | | | | 29* |
| PMADRL | PMADRL<7:0> | | | | | | | | 30 |
| PMADRH | — | — | — | — | — | — | PMADRH<1:0> | | 30 |
| PMDATL | PMDATL<7:0> | | | | | | | | 30 |
| PMDATH | — | — | PMDATH<5:0> | | | | | | 30 |
| INTCON | GIE | PEIE | TOIE | INTE | IOCF | TOIF | INTF | IOCF | 20 |

Legend: — = unimplemented location, read as '0'. Shaded cells are not used by Flash program memory module.

* Page provides register information.

TABLE 3-2: SUMMARY OF CONFIGURATION WORD WITH FLASH PROGRAM MEMORY

| Name | Bits | Bit -/7 | Bit -/6 | Bit 13/5 | Bit 12/4 | Bit 11/3 | Bit 10/2 | Bit 9/1 | Bit 8/0 | Register on Page |
|--------|------|---------|---------|----------|----------|----------|----------|------------|---------|------------------|
| CONFIG | 13:8 | — | — | DEBUG | CLKOUTEN | WRT<1:0> | | BOREN<1:0> | | 142 |
| | 7:0 | — | CP | MCLRE | PWRTE | WDTE | — | — | FOSC0 | |

Legend: — = unimplemented location, read as '1'. Shaded cells are not used by Flash program memory.

PIC12F752/HV752

NOTES:

4.0 OSCILLATOR MODULE

The internal oscillator module provides the following selectable system clock modes:

4.1 Overview

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

- 8 MHz (HFINTOSC)
- 4 MHz (HFINTOSC Postscaler)
- 1 MHz (HFINTOSC Postscaler)
- 31 kHz (LFINTOSC)

The oscillator module can be configured in one of two clock modes.

1. EC (external clock)
2. INTOSC (internal oscillator)

Clock Source modes are configured by the FOSC bit in the Configuration Word register (CONFIG).

FIGURE 4-1: PIC® MCU CLOCK SOURCE BLOCK DIAGRAM

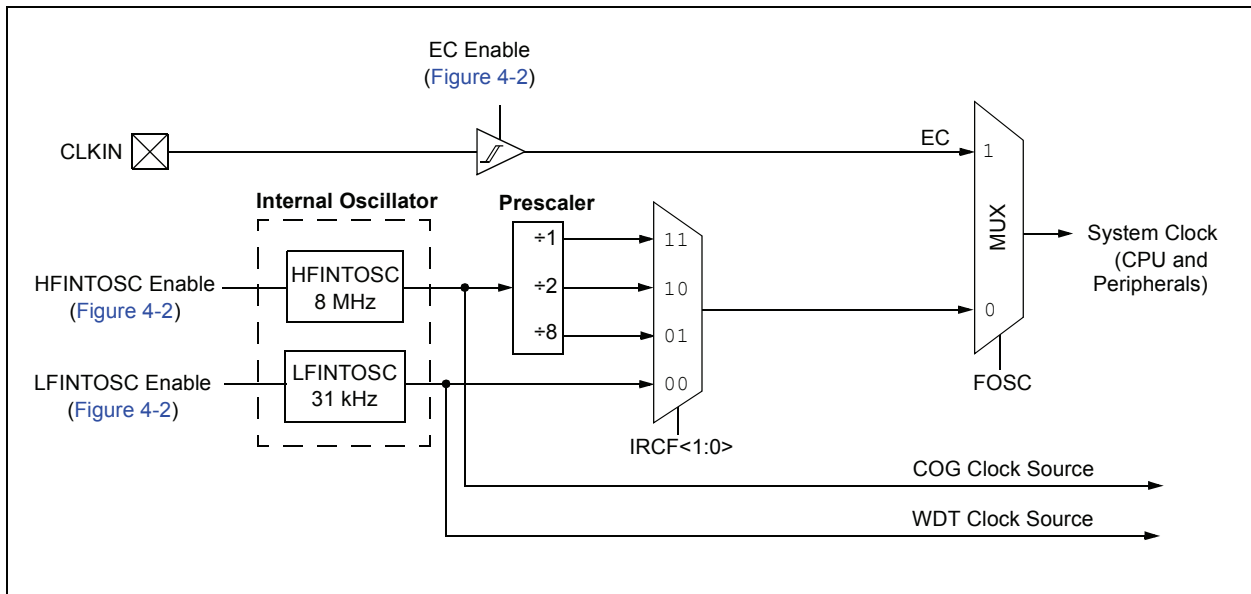
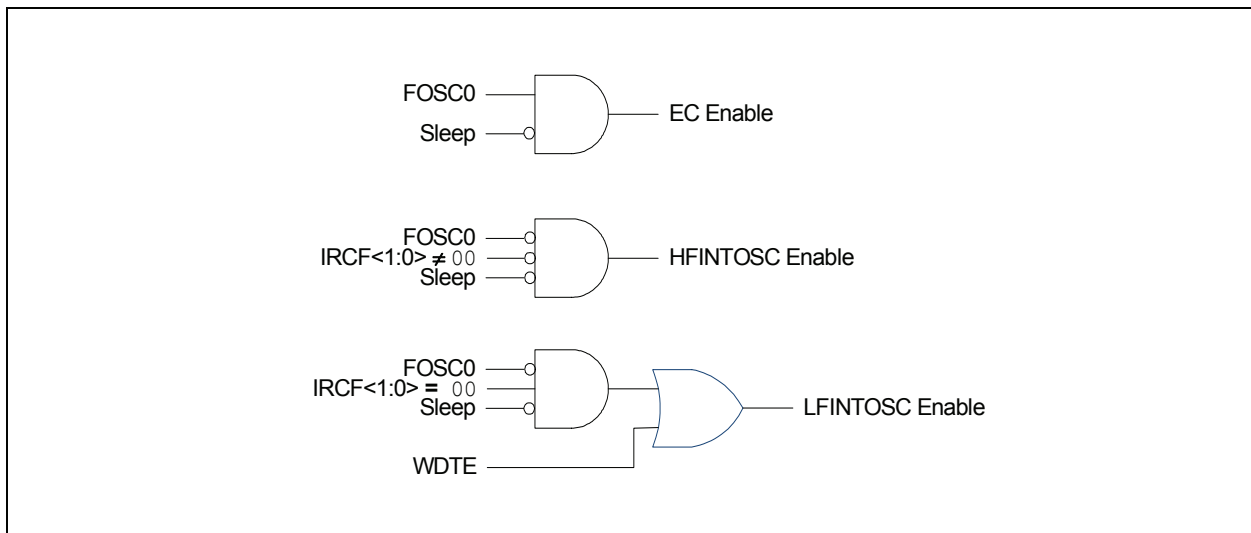


FIGURE 4-2: OSCILLATOR ENABLE



PIC12F752/HV752

4.2 Clock Source Modes

Clock Source modes can be classified as external or internal:

- The External Clock mode relies on an external clock for the clock source. For example, a clock module or clock output from another circuit.
- Internal clock sources are contained internally within the oscillator module. The oscillator module has four selectable clock frequencies:
 - 8 MHz
 - 4 MHz
 - 1 MHz
 - 31 kHz

The system clock can be selected between external or internal clock sources via the FOSC0 bit of the Configuration Word register (CONFIG).

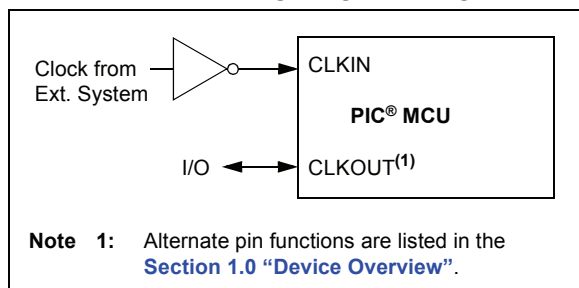
4.2.1 EC MODE

The External Clock (EC) mode allows an externally generated logic as the system clock source. The EC clock mode is selected when the FOSC0 bit of the Configuration Word is set.

When operating in this mode, an external clock source must be connected to the CLKIN input. The CLKOUT is available for either general purpose I/O or system clock output. [Figure 4-3](#) shows the pin connections for EC mode.

Because the PIC® MCU design is fully static, stopping the external clock input will have the effect of halting the device while leaving all data intact. Upon restarting the external clock, the device will resume operation as if no time had elapsed.

FIGURE 4-3: EXTERNAL CLOCK (EC) MODE OPERATION



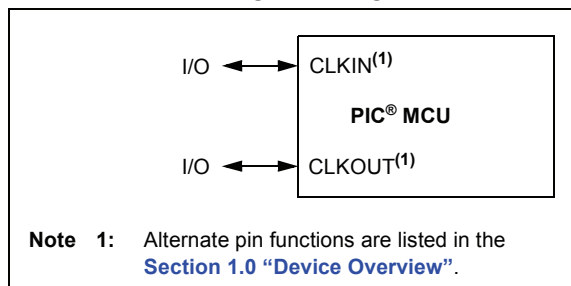
4.2.2 INTERNAL CLOCK MODE

Internal clock mode configures the internal oscillators as the system clock source. The internal clock mode is selected when the FOSC0 bit of the Configuration Word is cleared. The source and frequency are selected with the IRCF<1:0> bits of the OSCCON register.

When one of the HFINTOSC frequencies is selected, the frequency of the internal oscillator can be trimmed by adjusting the TUN<4:0> bits of the OSCTUNE register.

Operation after a Power-on Reset (POR) or wake-up from Sleep is delayed by the oscillator start-up time. Delays are typically longer for the LFINTOSC than HFINTOSC because of the very low-power operation and relatively narrow bandwidth of the LF internal oscillator. However, when another peripheral keeps the oscillator running during Sleep, the start-up time is delayed to allow the memory bias to stabilize.

FIGURE 4-4: INTERNAL CLOCK MODE OPERATION



4.2.2.1 Oscillator Ready Bits

The HTS and LTS bits of the OSCCON register indicate the status of the HFINTOSC and LFINTOSC, respectively. When either bit is set, it indicates that the corresponding oscillator is running and stable.

4.3 System Clock Output

The CLKOUT pin is available for general purpose I/O or system clock output. The $\overline{\text{CLKOUTEN}}$ bit of the Configuration Word controls the function of the CLKOUT pin.

When the $\overline{\text{CLKOUTEN}}$ bit is cleared, the CLKOUT pin is driven by the selected internal oscillator frequency divided by 4. The corresponding I/O pin always reads '0' in this configuration.

The CLKOUT signal may be used to provide a clock for external circuitry, synchronization, calibration, test or other application requirements.

When the $\overline{\text{CLKOUTEN}}$ bit is set, the system clock out function is disabled and the CLKOUT pin is available for general purpose I/O.

4.4 Oscillator Delay upon Wake-Up, Power-Up, and Base Frequency Change

In applications where the OSCTUNE register is used to shift the HFINTOSC frequency, the application should not expect the frequency to stabilize immediately. In this case, the frequency may shift gradually toward the new value. The time for this frequency shift is less than eight cycles of the base frequency.

A short delay is invoked upon power-up and when waking from sleep to allow the memory bias circuitry to stabilize. Table 4-1 shows examples where the oscillator delay is invoked.

TABLE 4-1: OSCILLATOR DELAY EXAMPLES

| Switch From | Switch To | Frequency | Oscillator Delay |
|-------------|-----------|-----------------|--|
| Sleep/POR | INTOSC | 31 kHz to 8 MHz | 10 μs internal delay to allow memory bias to stabilize. |
| Sleep/POR | EC | DC – 20 MHz | |

PIC12F752/HV752

4.5 Oscillator Control Registers

REGISTER 4-1: OSCCON: OSCILLATOR CONTROL REGISTER

| | | | | | | | |
|-------|-----|-----------|---------|-----|-------|-------|-------|
| U-0 | U-0 | R/W-0/u | R/W-1/u | U-0 | R-0/u | R-0/u | U-0 |
| — | — | IRCF<1:0> | | — | HTS | LTS | — |
| bit 7 | | | | | | | bit 0 |

Legend:

R = Readable bit

W = Writable bit

U = Unimplemented bit, read as '0'

-n = Value at POR

'1' = Bit is set

'0' = Bit is cleared

x = Bit is unknown

bit 7-6 **Unimplemented:** Read as '0'

bit 5-4 **IRCF<1:0>:** Internal Oscillator Frequency Select bits

11 = 8 MHz

10 = 4 MHz

01 = 1 MHz (Reset default)

00 = 31 kHz (LFINTOSC)

bit 3 **Unimplemented:** Read as '0'

bit 2 **HTS:** HFINTOSC Status bit

1 = HFINTOSC is stable

0 = HFINTOSC is not stable

bit 1 **LTS:** LFINTOSC Status bit

1 = LFINTOSC is stable

0 = LFINTOSC is not stable

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

4.5.1 OSCTUNE REGISTER

The oscillator is factory calibrated, but can be adjusted in software by writing to the OSCTUNE register ([Register 4-2](#)).

The default value of the OSCTUNE register is '0'. The value is a 5-bit two's complement number.

When the OSCTUNE register is modified, the frequency will begin shifting to the new frequency. Code execution continues during this shift. There is no indication that the shift has occurred.

REGISTER 4-2: OSCTUNE: OSCILLATOR TUNING REGISTER

| | | | | | | | |
|-------|-----|-----|----------|---------|---------|---------|---------|
| U-0 | U-0 | U-0 | R/W-0/u | R/W-0/u | R/W-0/u | R/W-0/u | R/W-0/u |
| — | — | — | TUN<4:0> | | | | |
| bit 7 | | | | | | | bit 0 |

Legend:

R = Readable bit W = Writable bit U = Unimplemented bit, read as '0'
 -n = Value at POR '1' = Bit is set '0' = Bit is cleared x = Bit is unknown

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

bit 4-0 **TUN<4:0>:** Frequency Tuning bits

- 01111 = Maximum frequency
- 01110 =
-
-
-
- 00001 =
- 00000 = Oscillator module is running at the calibrated frequency.
- 11111 =
-
-
-
- 10000 = Minimum frequency

TABLE 4-2: SUMMARY OF REGISTERS ASSOCIATED WITH CLOCK SOURCES

| Name | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | Register on Page |
|---------|-------|-------|-----------|----------|-------|-------|-------|-------|------------------|
| OSCCON | — | — | IRCF<1:0> | | — | HTS | LTS | — | 42 |
| OSCTUNE | — | — | — | TUN<4:0> | | | | | 43 |

Legend: x = unknown, u = unchanged, - = unimplemented locations read as '0'. Shaded cells are not used by oscillators.

- Note 1:** Other (non Power-up) Resets include $\overline{\text{MCLR}}$ Reset and Watchdog Timer Reset during normal operation.
2: See Configuration Word register ([Register 17-1](#)) for operation of all register bits.

TABLE 4-3: SUMMARY OF CONFIGURATION WORD CLOCK SOURCES

| Name | Bits | Bit -/7 | Bit -/6 | Bit 13/5 | Bit 12/4 | Bit 11/3 | Bit 10/2 | Bit 9/1 | Bit 8/0 | Register on Page |
|--------|------|---------|------------------------|----------------------------|------------------------------|----------|----------|------------|---------|------------------|
| CONFIG | 13:8 | — | — | $\overline{\text{DEBUG}}$ | $\overline{\text{CLKOUTEN}}$ | WRT<1:0> | | BOREN<1:0> | | 142 |
| | 7:0 | — | $\overline{\text{CP}}$ | $\overline{\text{MCLRRE}}$ | $\overline{\text{PWRTE}}$ | WDTE | — | — | FOSC0 | |

Legend: — = unimplemented location, read as '1'. Shaded cells are not used by oscillator module.

PIC12F752/HV752

NOTES:

5.0 I/O PORTS

For this device there is one port available, PORTA. In general, when a peripheral is enabled on a port pin, that pin cannot be used as a general purpose output. However, the pin can still be read.

PORTA has three standard registers for its operation. These registers are:

- TRISA registers (data direction)
- PORTA registers (reads the levels on the pins of the device)
- LATA registers (output latch)

Some ports may have one or more of the following additional registers. These registers are:

- ANSELA (analog select)
- WPUA (weak pull-up)

The Data Latch (LATA register) is useful for read-modify-write operations on the value that the I/O pins are driving.

A write operation to the LATA register has the same effect as a write to the corresponding PORTA register. A read of the LATA register reads the values held in the I/O PORT latches, while a read of the PORTA register reads the actual I/O pin value.

Ports that support analog inputs have an associated ANSEL register. When an ANSELA bit is set, the digital input buffer associated with that bit is disabled. Disabling the input buffer prevents analog signal levels on the pin between a logic high and low from causing excessive current in the logic input circuitry. A simplified model of a generic I/O port, without the interfaces to other peripherals, is shown in [Figure 5-1](#).

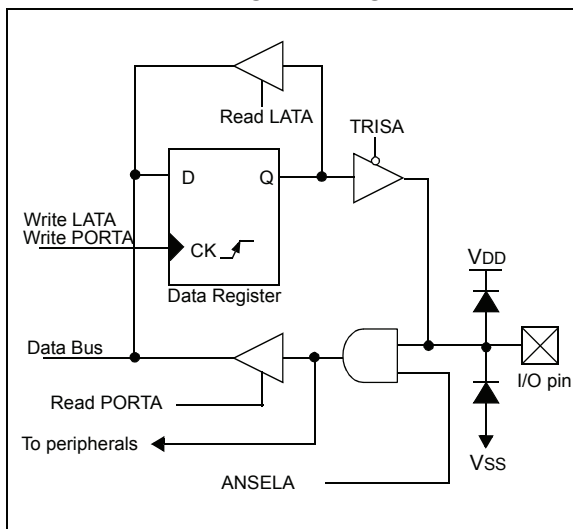
EXAMPLE 5-1: INITIALIZING PORTA

```

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

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

FIGURE 5-1: GENERIC I/O PORTA OPERATION



PIC12F752/HV752

5.1 Alternate Pin Function

The Alternate Pin Function Control (APFCON) register is used to steer specific peripheral input and output functions between different pins. The APFCON register is shown in [Register 5-1](#). For this device family, the following functions can be moved between different pins.

- Timer1 Gate
- COG1

These bits have no effect on the values of any TRIS register. PORT and TRIS overrides will be routed to the correct pin. The unselected pin will be unaffected.

5.2 Alternate Pin Function Control Register

REGISTER 5-1: APFCON: ALTERNATE PIN FUNCTION CONTROL REGISTER

| U-0 | U-0 | U-0 | R/W-0/0 | R/W-0/0 | R/W-0/0 | R/W-0/0 | R/W-0/0 |
|-------|-----|-----|---------|---------|----------|-----------|-----------|
| — | — | — | T1GSEL | — | COG1FSEL | COG1O1SEL | COG1O0SEL |
| bit 7 | | | | | | | bit 0 |

Legend:

R = Readable bit

W = Writable bit

U = Unimplemented bit, read as '0'

u = bit is unchanged

x = Bit is unknown

-n/n = Value at POR and BOR/Value at all other Resets

'1' = Bit is set

'0' = Bit is cleared

- bit 7-5 **Unimplemented:** Read as '0'.
- bit 4 **T1GSEL:** Timer 1 Gate Input Pin Selection bit
1 = T1G function is on RA3
0 = T1G function is on RA4
- bit 3 **Unimplemented:** Read as '0'.
- bit 2 **COG1FSEL:** COG1 Fault Input Pin Selection bit
1 = COG1FLT is on RA3
0 = COG1FLT is on RA4
- bit 1 **COG1O1SEL:** COG1 Output 1 Pin Selection bit
1 = COG1OUT1 is on RA4
0 = COG1OUT1 is on RA0
- bit 0 **COG1O0SEL:** COG1 Output 0 Pin Selection bit
1 = COG1OUT0 is on RA5
0 = COG1OUT0 is on RA2

5.3 PORTA and the TRISA Registers

PORTA is a 6-bit wide port with 5 bidirectional and 1 input-only pin. The corresponding data direction register is TRISA (Register 5-2). Setting a TRISA bit (= 1) will make the corresponding PORTA pin an input (i.e., disable the output driver). Clearing a TRISA bit (= 0) will make the corresponding PORTA pin an output (i.e., enables output driver and puts the contents of the output latch on the selected pin). The exception is RA3, which is input only and its TRIS bit will always read as '1'. Example 5-1 shows how to initialize PORTA.

Reading the PORTA register (Register 5-2) reads the status of the pins, whereas writing to it will write to the PORT latch. All write operations are read-modify-write operations. Therefore, a write to a port implies that the port pins are read, this value is modified and then written to the PORT data latch. RA3 reads '0' when MCLR = 1.

The TRISA register controls the direction of the PORTA pins, even when they are being used as analog inputs. The user must ensure the bits in the TRISA register are maintained set when using them as analog inputs. I/O pins configured as analog input always read '0'.

Note: The ANSEL register must be initialized to configure an analog channel as a digital input. Pins configured as analog inputs will read '0' and cannot generate an interrupt.

5.3.1 PORTA FUNCTIONS AND OUTPUT PRIORITIES

Each PORTA pin is multiplexed with other functions. The pins, their combined functions and their output priorities are shown in Table 5-1.

When multiple outputs are enabled, the actual pin control goes to the peripheral with the highest priority.

Analog input functions, such as comparator inputs, are not shown in the priority lists. These inputs are active when the peripheral is enabled and the input multiplexer for the pin is selected. The Analog mode, set with the ANSELA register, disables the digital input buffer thereby preventing excessive input current when the analog input voltage is between logic states. Digital output functions may control the pin when it is in Analog mode with the priority shown in Table 5-1.

TABLE 5-1: PORTA OUTPUT PRIORITY

| Pin Name | Function Priority ⁽¹⁾ |
|----------|---|
| RA0 | ICSPDAT REFOUT DACOUT COG1OUT1 ⁽²⁾ RA0 |
| RA1 | RA1 |
| RA2 | COG1OUT0 ⁽²⁾ C1OUT C2OUT CCP1 RA2 |
| RA3 | None |
| RA4 | CLKOUT COG1OUT1 ⁽³⁾ RA4 |
| RA5 | COG1OUT0 ⁽³⁾ RA5 |

- Note**
- 1: Priority listed from highest to lowest.
 - 2: Default function pin (see APFCON register).
 - 3: Alternate function pin (see APFCON register).

PIC12F752/HV752

5.4 PORTA Control Registers

REGISTER 5-2: PORTA: PORTA REGISTER

| | | | | | | | |
|-------|-----|---------|---------|-------|---------|---------|---------|
| U-0 | U-0 | R/W-x/u | R/W-x/u | R-x/x | R/W-x/u | R/W-x/u | R/W-x/u |
| — | — | RA5 | RA4 | RA3 | RA2 | RA1 | RA0 |
| bit 7 | | | | | | | bit 0 |

Legend:

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

bit 7-6 **Unimplemented:** Read as '0'
bit 5-0 **RA<5:0>:** PORTA I/O Value bits⁽¹⁾
 1 = Port pin is ≥ V_{IH}
 0 = Port pin is ≤ V_{IL}

Note 1: Writes to PORTA are actually written to corresponding LATA register. Reads from PORTA register is return of actual I/O pin values.

REGISTER 5-3: TRISA: PORTA TRI-STATE REGISTER

| | | | | | | | |
|-------|-----|---------|---------|-----------------------|---------|---------|---------|
| U-0 | U-0 | R/W-1/1 | R/W-1/1 | R-1/1 | R/W-1/1 | R/W-1/1 | R/W-1/1 |
| — | — | TRISA5 | TRISA4 | TRISA3 ⁽¹⁾ | TRISA2 | TRISA1 | TRISA0 |
| bit 7 | | | | | | | bit 0 |

Legend:

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

bit 7-6 **Unimplemented:** Read as '0'
bit 5-0 **TRISA<5:0>:** PORTA Tri-State Control bits⁽¹⁾
 1 = PORTA pin configured as an input (tri-stated)
 0 = PORTA pin configured as an output

Note 1: TRISA3 always reads '1'.

REGISTER 5-4: LATA: PORTA DATA LATCH REGISTER

| | | | | | | | |
|-------|-----|---------|---------|-----|---------|---------|---------|
| U-0 | U-0 | R/W-x/u | R/W-x/u | U-0 | R/W-x/u | R/W-x/u | R/W-x/u |
| — | — | LATA5 | LATA4 | — | LATA2 | LATA1 | LATA0 |
| bit 7 | | | | | | | bit 0 |

Legend:

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

bit 7-6 **Unimplemented:** Read as '0'
bit 5-4 **LATA<5:4>:** PORTA Output Latch Value bits⁽¹⁾
bit 3 **Unimplemented:** Read as '0'
bit 2-0 **LATA<2:0>:** PORTA Output Latch Value bits⁽¹⁾

Note 1: Writes to PORTA are actually written to corresponding LATA register. Reads from PORTA register is return of actual I/O pin values.

5.5 Additional Pin Functions

Every PORTA pin on the PIC12F752 has an interrupt-on-change option and a weak pull-up option. The next three sections describe these functions.

5.5.1 ANSELA REGISTER

The ANSELA register ([Register 5-5](#)) is used to configure the Input mode of an I/O pin to analog. Setting the appropriate ANSELA bit high will cause all digital reads on the pin to be read as '0' and allow analog functions on the pin to operate correctly.

The state of the ANSELA bits has no effect on digital output functions. A pin with TRIS clear and ANSEL set will still operate as a digital output, but the Input mode will be analog. This can cause unexpected behavior when executing read-modify-write instructions on the affected port.

Note: The ANSELA bits default to the Analog mode after Reset. To use any pins as digital general purpose or peripheral inputs, the corresponding ANSEL bits must be initialized to '0' by user software.

5.5.2 WEAK PULL-UPS

Each of the PORTA pins, except RA3, has an individually configurable internal weak pull-up. Control bits WPUx enable or disable each pull-up. Refer to [Register 5-6](#). Each weak pull-up is automatically turned off when the port pin is configured as an output. The pull-ups are disabled on a Power-on Reset by the RAPU bit of the OPTION_REG register). A weak pull-up is automatically enabled for RA3 when configured as $\overline{\text{MCLR}}$ and disabled when RA3 is an I/O. There is no software control of the $\overline{\text{MCLR}}$ pull-up.

5.5.3 INTERRUPT-ON-CHANGE

Each PORTA pin is individually configurable as an interrupt-on-change pin. Control bits IOCA enable or disable the interrupt function for each pin. Refer to [Register 5-8](#). The interrupt-on-change is disabled on a Power-on Reset.

For enabled interrupt-on-change pins, the values are compared with the old value latched on the last read of PORTA. The 'mismatch' outputs of the last read are OR'd together to set the PORTA Change Interrupt Flag bit (IOCIF) in the INTCON register ([Register 2-3](#)).

This interrupt can wake the device from Sleep. The user, in the Interrupt Service Routine, clears the interrupt by:

- a) Any read of PORTA AND Clear flag bit IOCIF.
This will end the mismatch condition;
- OR
- b) Any write of PORTA AND Clear flag bit IOCIF
will end the mismatch condition;

A mismatch condition will continue to set flag bit IOCIF. Reading PORTA will end the mismatch condition and allow flag bit IOCIF to be cleared. The latch holding the last read value is not affected by a $\overline{\text{MCLR}}$ nor BOR Reset. After these Resets, the IOCIF flag will continue to be set if a mismatch is present.

Note: If a change on the I/O pin should occur when any PORTA operation is being executed, then the IOCIF interrupt flag may not get set.

5.5.4 SLEW RATE CONTROL

Two of the PORTA pins, RA0 and RA2, are equipped with high current driver circuitry. The SLRCONA register provides reduced slew rate control to mitigate possible EMI radiation from these pins.

PIC12F752/HV752

REGISTER 5-5: ANSELA: PORTA ANALOG SELECT REGISTER

| | | | | | | | |
|-------|-----|-------|-------|-----|-------|-------|-------|
| U-0 | U-0 | R/W-1 | R/W-1 | U-0 | R/W-1 | R/W-1 | R/W-1 |
| — | — | ANSA5 | ANSA4 | — | ANSA2 | ANSA1 | ANSA0 |
| bit 7 | | | | | | | bit 0 |

Legend:

R = Readable bit W = Writable bit U = Unimplemented bit, read as '0'
 -n = Value at POR '1' = Bit is set '0' = Bit is cleared x = Bit is unknown

- bit 7-6 **Unimplemented:** Read as '0'
- bit 5-4 **ANSA<5:4>:** Analog Select Between Analog or Digital Function on Pin RA<5:4> bits
 1 = Analog input. Pin is assigned as analog input⁽¹⁾.
 0 = Digital I/O. Pin is assigned to port or special function.
- bit 3 **Unimplemented:** Read as '0'
- bit 2-0 **ANSA<2:0>** Analog Select Between Analog or Digital Function on Pin RA<2:0> bits
 1 = Analog input. Pin is assigned as analog input.⁽¹⁾
 0 = Digital I/O. Pin is assigned to port or special function.

Note 1: Setting a pin to an analog input automatically disables the digital input circuitry, weak pull-ups, and interrupt-on-change if available. The corresponding TRIS bit must be set to Input mode in order to allow external control of the voltage on the pin.

REGISTER 5-6: WPUA: WEAK PULL-UP PORTA REGISTER

| | | | | | | | |
|-------|-----|-------|-------|-------|-------|-------|-------|
| U-0 | U-0 | R/W-1 | R/W-1 | R/W-1 | R/W-1 | R/W-1 | R/W-1 |
| — | — | WPU5 | WPU4 | WPU3 | WPU2 | WPU1 | WPU0 |
| bit 7 | | | | | | | bit 0 |

Legend:

R = Readable bit W = Writable bit U = Unimplemented bit, read as '0'
 -n = Value at POR '1' = Bit is set '0' = Bit is cleared x = Bit is unknown

- bit 7-6 **Unimplemented:** Read as '0'
- bit 5-0 **WPU<5:0>:** Weak Pull-up Control bits
 1 = Pull-up enabled
 0 = Pull-up disabled

Note 1: Global $\overline{\text{RAPU}}$ must be enabled for individual pull-ups to be enabled.
Note 2: The weak pull-up device is automatically disabled if the pin is in Output mode (TRISA = 0).
Note 3: The RA3 pull-up is enabled when configured as $\overline{\text{MCLR}}$ in the Configuration Word, otherwise it is disabled as an input and reads as '0'.

REGISTER 5-7: SLRCONA: SLEW RATE CONTROL REGISTER

| | | | | | | | |
|-------|-----|-----|-----|-----|-------|-------|-------|
| U-0 | U-0 | U-0 | U-0 | U-0 | R/W-0 | U-0 | R/W-0 |
| — | — | — | — | — | SLRA2 | — | SLRA0 |
| bit 7 | | | | | | bit 0 | |

Legend:

R = Readable bit

W = Writable bit

U = Unimplemented bit, read as '0'

-n = Value at POR

'1' = Bit is set

'0' = Bit is cleared

x = Bit is unknown

bit 7-3 **Unimplemented:** Read as '0'

bit 2 **SLRA2:** Slew Rate Control bit
 1 = Pin voltage slews at limited rate
 0 = Pin voltage slews at maximum rate

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

bit 0 **SLRA0:** Slew Rate Control bit
 1 = Pin voltage slews at limited rate
 0 = Pin voltage slews at maximum rate

- Note 1:** Global $\overline{\text{RAPU}}$ must be enabled for individual pull-ups to be enabled.
- 2:** The weak pull-up device is automatically disabled if the pin is in Output mode ($\text{TRISA} = 0$).
- 3:** The RA3 pull-up is enabled when configured as $\overline{\text{MCLR}}$ in the Configuration Word, otherwise it is disabled as an input and reads as '0'.

PIC12F752/HV752

REGISTER 5-8: IOCAP: INTERRUPT-ON-CHANGE POSITIVE EDGE REGISTER

| | | | | | | | |
|-------|-----|---------|---------|---------|---------|---------|---------|
| U-0 | U-0 | R/W-0/0 | R/W-0/0 | R/W-0/0 | R/W-0/0 | R/W-0/0 | R/W-0/0 |
| — | — | IOCAP5 | IOCAP4 | IOCAP3 | IOCAP2 | IOCAP1 | IOCAP0 |
| bit 7 | | | | | | | bit 0 |

Legend:

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

bit 7-6 **Unimplemented:** Read as '0'

bit 5-0 **IOCAP<5:0>:** Interrupt-on-Change Positive Edge Enable bits

- 1 = Interrupt-on-Change enabled on the pin for a positive going edge. Associated Status bit and interrupt flag will be set upon detecting an edge.
- 0 = Interrupt-on-Change disabled for the associated pin.

REGISTER 5-9: IOCAN: INTERRUPT-ON-CHANGE NEGATIVE EDGE REGISTER

| | | | | | | | |
|-------|-----|---------|---------|---------|---------|---------|---------|
| U-0 | U-0 | R/W-0/0 | R/W-0/0 | R/W-0/0 | R/W-0/0 | R/W-0/0 | R/W-0/0 |
| — | — | IOCAN5 | IOCAN4 | IOCAN3 | IOCAN2 | IOCAN1 | IOCAN0 |
| bit 7 | | | | | | | bit 0 |

Legend:

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

bit 7-6 **Unimplemented:** Read as '0'

bit 5-0 **IOCAN<5:0>:** Interrupt-on-Change Negative Edge Enable bits

- 1 = Interrupt-on-Change enabled on the pin for a negative going edge. Associated Status bit and interrupt flag will be set upon detecting an edge.
- 0 = Interrupt-on-Change disabled for the associated pin.

REGISTER 5-10: IOCAF: INTERRUPT-ON-CHANGE FLAG REGISTER

| | | | | | | | |
|-------|-----|------------|------------|------------|------------|------------|------------|
| U-0 | U-0 | R/W/HS-0/0 | R/W/HS-0/0 | R/W/HS-0/0 | R/W/HS-0/0 | R/W/HS-0/0 | R/W/HS-0/0 |
| — | — | IOCAF5 | IOCAF4 | IOCAF3 | IOCAF2 | IOCAF1 | IOCAF0 |
| bit 7 | | | | | | | bit 0 |

Legend:

R = Readable bit W = Writable bit U = Unimplemented bit, read as '0'
u = Bit is unchanged x = Bit is unknown -n/n = Value at POR and BOR/Value at all other Resets
'1' = Bit is set '0' = Bit is cleared HS - Bit is set in hardware

bit 7-6 **Unimplemented:** Read as '0'

bit 5-0 **IOCAF<5:0>:** Interrupt-on-Change Flag bits

- 1 = An enabled change was detected on the associated pin.
Set when IOCAPx = 1 and a rising edge was detected on RBx, or when IOCANx = 1 and a falling edge was detected on RAx.
- 0 = No change was detected, or the user cleared the detected change.

TABLE 5-2: SUMMARY OF REGISTERS ASSOCIATED WITH PORTA

| Name | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | Register on Page |
|------------|--------|-----------|------------|--------|-----------------------|----------|-----------|-----------|------------------|
| ADCON0 | ADFM | VCFG | CHS<3:0> | | | | GO/DONE | ADON | 106 |
| ADCON1 | — | ADCS<2:0> | | | — | — | — | — | 107 |
| ANSELA | — | — | ANSA5 | ANSA4 | — | ANSA2 | ANSA1 | ANSA0 | 50 |
| APFCON | — | — | — | T1GSEL | — | COG1FSEL | COG1O1SEL | COG1O0SEL | 46 |
| CM1CON0 | C1ON | C1OUT | C1OE | C1POL | C1ZLF | C1SP | C1HYS | C1SYNC | 127 |
| CM2CON0 | C2ON | C2OUT | C2OE | C2POL | C2ZLF | C2SP | C2HYS | C2SYNC | 127 |
| CM1CON1 | C1INTP | C1INTN | C1PCH<1:0> | | — | — | — | C1NCH0 | 128 |
| CM2CON1 | C2NTP | C2INTN | C2PCH<1:0> | | — | — | — | C2NCH0 | 128 |
| DACCON0 | DACEN | DACRNG | DACOE | — | — | DACPSS0 | — | — | 118 |
| IOCAF | — | — | IOCAF5 | IOCAF4 | IOCAF3 | IOCAF2 | IOCAF1 | IOCAF0 | 52 |
| IOCAN | — | — | IOCAN5 | IOCAN4 | IOCAN3 | IOCAN2 | IOCAN1 | IOCAN0 | 52 |
| IOCAP | — | — | IOCAP5 | IOCAP4 | IOCAP3 | IOCAP2 | IOCAP1 | IOCAP0 | 52 |
| LATA | — | — | LATA5 | LATA4 | — | LATA2 | LATA1 | LATA0 | 48 |
| OPTION_REG | RAPU | INTEDG | T0CS | T0SE | PSA | PS<2:0> | | | 19 |
| PORTA | — | — | RA5 | RA4 | RA3 | RA2 | RA1 | RA0 | 48 |
| SLRCONA | — | — | — | — | — | SLRA2 | — | SLRA0 | 51 |
| TRISA | — | — | TRISA5 | TRISA4 | TRISA3 ⁽¹⁾ | TRISA2 | TRISA1 | TRISA0 | 48 |

Legend: x = unknown, u = unchanged, — = unimplemented locations read as '0'. Shaded cells are not used by PORTA.

Note 1: TRISA3 always reads '1'.

PIC12F752/HV752

NOTES:

6.0 TIMER0 MODULE

The Timer0 module is an 8-bit timer/counter with the following features:

- 8-bit timer/counter register (TMR0)
- 8-bit prescaler (shared with Watchdog Timer)
- Programmable internal or external clock source
- Programmable external clock edge selection
- Interrupt on overflow

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

6.1 Timer0 Operation

When used as a timer, the Timer0 module can be used as either an 8-bit timer or an 8-bit counter.

6.1.1 8-BIT TIMER MODE

When used as a timer, the Timer0 module will increment every instruction cycle (without prescaler). Timer mode is selected by clearing the T0CS bit of the OPTION register to '0'.

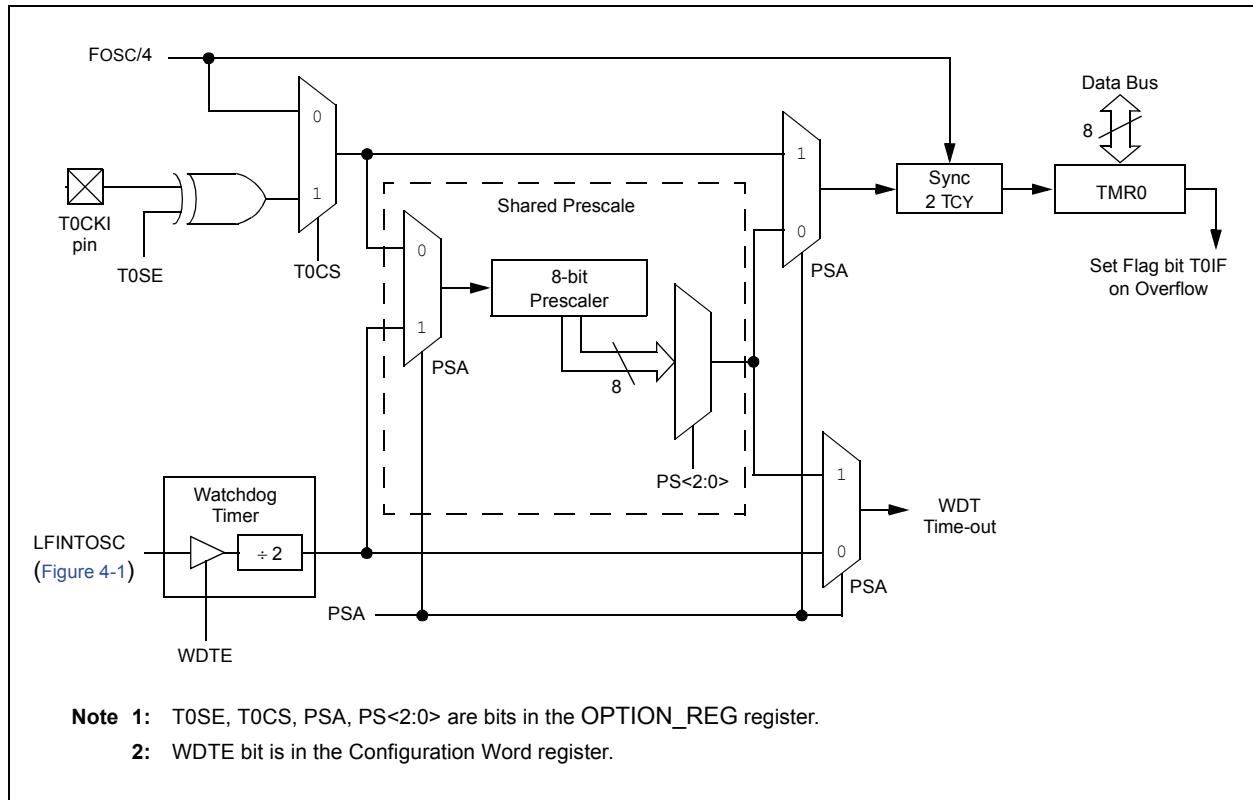
When TMR0 is written, the increment is inhibited for two instruction cycles immediately following the write.

Note: The value written to the TMR0 register can be adjusted, in order to account for the two instruction cycle delay when TMR0 is written.

6.1.2 8-BIT COUNTER MODE

When used as a counter, the Timer0 module will increment on every rising or falling edge of the T0CKI pin. The incrementing edge is determined by the T0SE bit of the OPTION_REG register. Counter mode is selected by setting the T0CS bit of the OPTION register to '1'.

FIGURE 6-1: TIMER0 WITH SHARED PRESCALE BLOCK DIAGRAM



PIC12F752/HV752

6.1.3 SOFTWARE PROGRAMMABLE PRESCALER

A single software programmable prescaler is available for use with either Timer0 or the Watchdog Timer (WDT), but not both simultaneously. The prescaler assignment is controlled by the PSA bit of the OPTION register. To assign the prescaler to Timer0, the PSA bit must be cleared to a '0'.

There are 8 prescaler options for the Timer0 module ranging from 1:2 to 1:256. The prescale values are selectable via the PS<2:0> bits of the OPTION register. In order to have a 1:1 prescaler value for the Timer0 module, the prescaler must be assigned to the WDT module.

The prescaler is not readable or writable. When assigned to the Timer0 module, all instructions writing to the TMR0 register will clear the prescaler.

When the prescaler is assigned to WDT, (PSA = 1), a CLRWDT instruction will clear the prescaler along with the WDT.

6.1.3.1 Switching Prescaler Between Timer0 and WDT Modules

As a result of having the prescaler assigned to either Timer0 or the WDT, it is possible to generate an unintended device Reset when switching prescaler values. When changing the prescaler assignment from Timer0 to the WDT module, the instruction sequence shown in [Example 6-1](#), must be executed.

EXAMPLE 6-1: CHANGING PRESCALER (TIMER0 → WDT)

```
BANKSEL TMR0      ;
CLRWDT            ;Clear WDT
CLRFB   TMR0      ;Clear TMR0 and
                  ;prescaler
BANKSEL OPTION_REG ;
BSF   OPTION_REG,PSA ;Select WDT
CLRWDT            ;
                  ;
MOVLW  b'11111000' ;Mask prescaler
ANDWF  OPTION_REG,W ;bits
IORLW  b'00000101' ;Set WDT prescaler
MOVWF  OPTION_REG  ;to 1:32
```

When changing the prescaler assignment from the WDT to the Timer0 module, the following instruction sequence must be executed (see [Example 6-2](#)).

EXAMPLE 6-2: CHANGING PRESCALER (WDT → TIMER0)

```
CLRWDT            ;Clear WDT and
                  ;prescaler
BANKSEL OPTION_REG ;
MOVLW  b'11110000' ;Mask TMR0 select and
ANDWF  OPTION_REG,W ;prescaler bits
IORLW  b'00000011' ;Set prescale to 1:16
MOVWF  OPTION_REG  ;
```

6.1.4 TIMER0 INTERRUPT

Timer0 will generate an interrupt when the TMR0 register overflows from FFh to 00h. The T0IF interrupt flag bit of the INTCON register is set every time the TMR0 register overflows, regardless of whether or not the Timer0 interrupt is enabled. The T0IF bit must be cleared in software. The Timer0 interrupt enable is the T0IE bit of the INTCON register.

Note: The Timer0 interrupt cannot wake the processor from Sleep since the timer is frozen during Sleep.

6.1.5 USING TIMER0 WITH AN EXTERNAL CLOCK

When Timer0 is in Counter mode, the synchronization of the T0CKI input and the Timer0 register is accomplished by sampling the prescaler output on the Q2 and Q4 cycles of the internal phase clocks. Therefore, the high and low periods of the external clock source must meet the timing requirements as shown in [Section 20.0 "Electrical Specifications"](#).

6.2 Option and Timer0 Control Register

REGISTER 6-1: OPTION_REG: OPTION REGISTER

| | | | | | | | |
|--------------------------|--------|-------|-------|-------|---------|-------|-------|
| R/W-1 | R/W-1 | R/W-1 | R/W-1 | R/W-1 | R/W-1 | R/W-1 | R/W-1 |
| $\overline{\text{RAPU}}$ | INTEDG | T0CS | T0SE | PSA | PS<2:0> | | |
| bit 7 | | | | | | | bit 0 |

Legend:

R = Readable bit W = Writable bit U = Unimplemented bit, read as '0'
 -n = Value at POR '1' = Bit is set '0' = Bit is cleared x = Bit is unknown

- bit 7 **$\overline{\text{RAPU}}$** : PORTA Pull-up Enable bit
 1 = PORTA pull-ups are disabled
 0 = PORTA pull-ups are enabled by individual PORT latch values in WPU register
- bit 6 **INTEDG**: Interrupt Edge Select bit
 1 = Interrupt on rising edge of INT pin
 0 = Interrupt on falling edge of INT pin
- bit 5 **T0CS**: TMR0 Clock Source Select bit
 1 = Transition on T0CKI pin
 0 = Internal instruction cycle clock (Fosc/4)
- bit 4 **T0SE**: TMR0 Source Edge Select bit
 1 = Increment on high-to-low transition on T0CKI pin
 0 = Increment on low-to-high transition on T0CKI pin
- bit 3 **PSA**: Prescaler Assignment bit
 1 = Prescaler is assigned to the WDT
 0 = Prescaler is assigned to the Timer0 module
- bit 2-0 **PS<2:0>**: Prescaler Rate Select bits

| Bit Value | TMR0 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 |

TABLE 6-1: SUMMARY OF REGISTERS ASSOCIATED WITH TIMER0

| Name | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | Register on Page |
|------------|--|--------|--------|--------|-----------------------|---------|--------|--------|------------------|
| TMR0 | Holding Register for the 8-bit Timer0 Register | | | | | | | | 55* |
| INTCON | GIE | PEIE | T0IE | INTE | IOCFIE | T0IF | INTF | IOCF | 20 |
| OPTION_REG | $\overline{\text{RAPU}}$ | INTEDG | T0CS | T0SE | PSA | PS<2:0> | | | 57 |
| TRISA | — | — | TRISA5 | TRISA4 | TRISA3 ⁽¹⁾ | TRISA2 | TRISA1 | TRISA0 | 48 |

Legend: — = Unimplemented locations, read as '0', u = unchanged, x = unknown. Shaded cells are not used by the Timer0 module.

* Page provides register information.

Note 1: TRISA3 always reads '1'.

PIC12F752/HV752

NOTES:

7.0 TIMER1 MODULE WITH GATE CONTROL

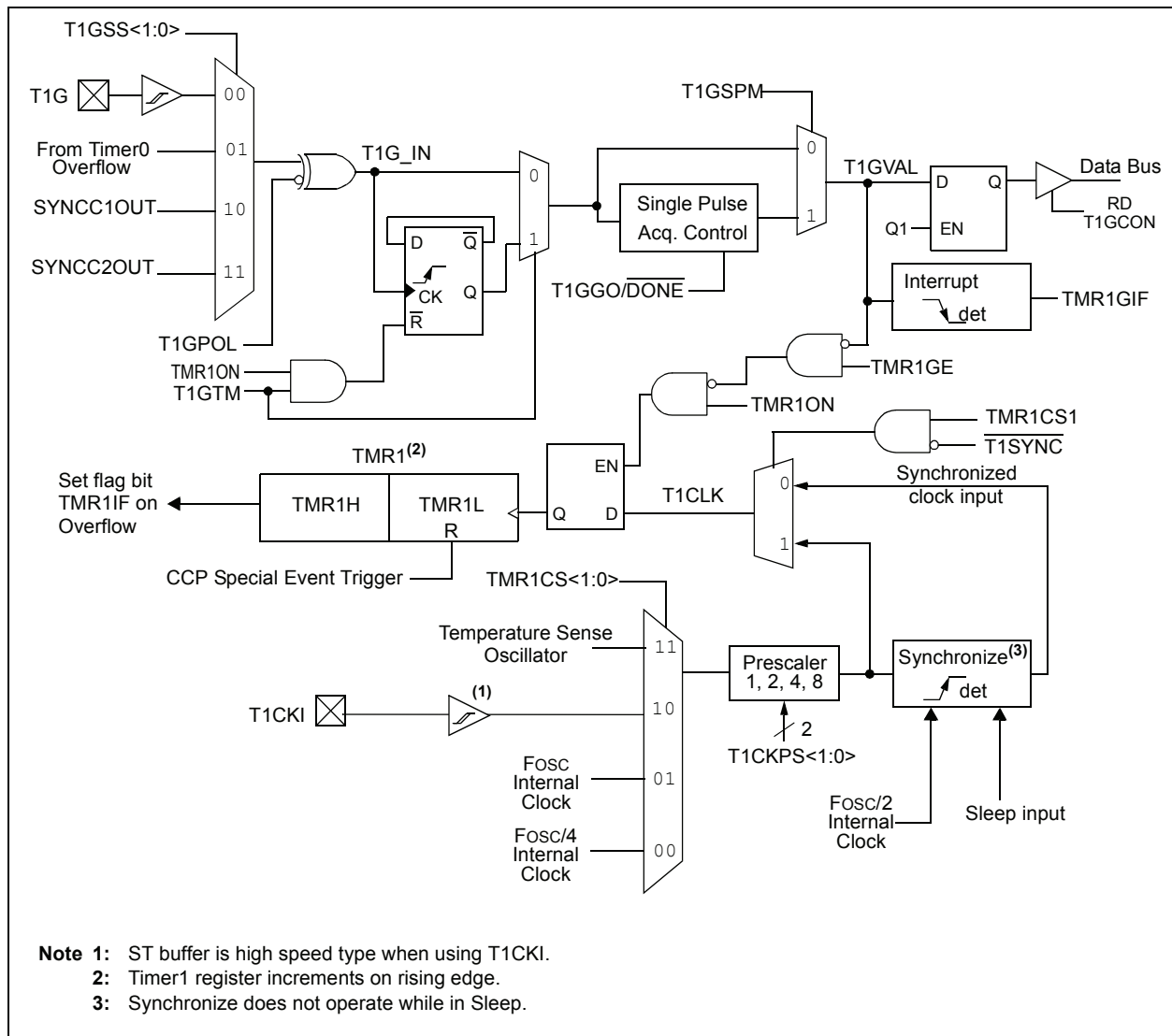
The Timer1 module is a 16-bit timer/counter with the following features:

- 16-bit timer/counter register pair (TMR1H:TMR1L)
- Selectable internal or external clock sources
- 2-bit prescaler
- Synchronous or asynchronous operation
- Multiple Timer1 gate (count enable) sources
- Interrupt on overflow
- Wake-up on overflow (external clock, Asynchronous mode only)
- Time base for the Capture/Compare function
- Special Event Trigger (with CCP)
- Selectable Gate Source Polarity

- Gate Toggle mode
- Gate Single-pulse mode
- Gate Value Status
- Gate Event Interrupt

Figure 7-1 is a block diagram of the Timer1 module.

FIGURE 7-1: TIMER1 BLOCK DIAGRAM



PIC12F752/HV752

7.1 Timer1 Operation

The Timer1 module is a 16-bit incrementing counter which is accessed through the TMR1H:TMR1L register pair. Writes to TMR1H or TMR1L directly update the counter.

When used with an internal clock source, the module is a timer and increments on every instruction cycle. When used with an external clock source, the module can be used as either a timer or counter and increments on every selected edge of the external source.

Timer1 is enabled by configuring the TMR1ON and TMR1GE bits in the T1CON and T1GCON registers, respectively. Table 7-1 displays the Timer1 enable selections.

TABLE 7-1: TIMER1 ENABLE SELECTIONS

| TMR1ON | TMR1GE | Timer1 Operation |
|--------|--------|------------------|
| 0 | 0 | Off |
| 0 | 1 | Off |
| 1 | 0 | Always On |
| 1 | 1 | Count Enabled |

7.2 Clock Source Selection

The TMR1CS<1:0> bits of the T1CON register are used to select the clock source for Timer1. Table 7-2 displays the clock source selections.

TABLE 7-2: CLOCK SOURCE SELECTIONS

| TMR1CS<1:0> | Clock Source |
|-------------|--------------------------------|
| 11 | Temperature Sense Oscillator |
| 10 | External Clocking on T1CKI Pin |
| 01 | System Clock (Fosc) |
| 00 | Instruction Clock (Fosc/4) |

7.2.1 INTERNAL CLOCK SOURCE

When the internal clock source is selected, the TMR1H:TMR1L register pair will increment on multiples of Fosc or Fosc/4 as determined by the Timer1 prescaler.

7.2.2 EXTERNAL CLOCK SOURCE

When the external clock source is selected, the Timer1 module may work as a timer or a counter. When enabled to count, Timer1 is incremented on the rising edge of the external clock input T1CKI.

Note: In Counter mode, a falling edge must be registered by the counter prior to the first incrementing rising edge (see Figure 7-2) after any one or more of the following conditions:

- Timer1 enabled after POR Reset
- Write to TMR1H or TMR1L
- Timer1 is disabled
- Timer1 is disabled (TMR1ON = 0) when T1CKI is high then Timer1 is enabled (TMR1ON=1) when T1CKI is low.

7.2.3 TEMPERATURE SENSE OSCILLATOR

When the Temperature Sense Oscillator source is selected, the TMR1H:TMR1L register pair will increment on multiples of the Temperature Sense Oscillator as determined by the Timer1 prescaler. The Temperature Sense Oscillator operates at 16 kHz typical.

7.3 Timer1 Prescaler

Timer1 has four prescaler options allowing 1, 2, 4 or 8 divisions of the clock input. The T1CKPS bits of the T1CON register control the prescale counter. The prescale counter is not directly readable or writable; however, the prescaler counter is cleared upon a write to TMR1H or TMR1L.

7.4 Timer1 Operation in Asynchronous Counter Mode

If control bit $\overline{T1SYNC}$ of the T1CON register is set, the external clock input is not synchronized. The timer increments asynchronously to the internal phase clocks. If external clock source is selected then the timer will continue to run during Sleep and can generate an interrupt on overflow, which will wake-up the processor. However, special precautions in software are needed to read/write the timer (see [Section 7.4.1 “Reading and Writing Timer1 in Asynchronous Counter Mode”](#)).

Note: When switching from synchronous to asynchronous operation, it is possible to skip an increment. When switching from asynchronous to synchronous operation, it is possible to produce an additional increment.

7.4.1 READING AND WRITING TIMER1 IN ASYNCHRONOUS COUNTER MODE

Reading TMR1H or TMR1L while the timer is running from an external asynchronous clock will ensure a valid read (taken care of in hardware). However, the user should keep in mind that reading the 16-bit timer in two 8-bit values itself, poses certain problems, since the timer may overflow between the reads.

For writes, it is recommended that the user simply stop the timer and write the desired values. A write contention may occur by writing to the timer registers, while the register is incrementing. This may produce an unpredictable value in the TMR1H:TMR1L register pair.

7.5 Timer1 Gate

Timer1 can be configured to count freely or the count can be enabled and disabled using Timer1 gate circuitry. This is also referred to as Timer1 gate count enable.

Timer1 gate can also be driven by multiple selectable sources.

7.5.1 TIMER1 GATE COUNT ENABLE

The Timer1 gate is enabled by setting the TMR1GE bit of the T1GCON register. The polarity of the Timer1 gate is configured using the T1GPOL bit of the T1GCON register.

When Timer1 Gate (T1G) input is active, Timer1 will increment on the rising edge of the Timer1 clock source. When Timer1 gate input is inactive, no incrementing will occur and Timer1 will hold the current count. See [Figure 7-3](#) for timing details.

TABLE 7-3: TIMER1 GATE ENABLE SELECTIONS

| T1CLK | T1GPOL | T1G | Timer1 Operation |
|-------|--------|-----|------------------|
| ↑ | 0 | 0 | Counts |
| ↑ | 0 | 1 | Holds Count |
| ↑ | 1 | 0 | Holds Count |
| ↑ | 1 | 1 | Counts |

7.5.2 TIMER1 GATE SOURCE SELECTION

The Timer1 gate source can be selected from one of four different sources. Source selection is controlled by the T1GSS bits of the T1GCON register. The polarity for each available source is also selectable. Polarity selection is controlled by the T1GPOL bit of the T1GCON register.

TABLE 7-4: TIMER1 GATE SOURCES

| T1GSS | Timer1 Gate Source |
|-------|---|
| 11 | SYNCC2OUT |
| 10 | SYNCC1OUT |
| 01 | Overflow of Timer0 (TMR0 increments from FFh to 00h) |
| 00 | Timer1 Gate Pin |

PIC12F752/HV752

7.5.2.1 T1G Pin Gate Operation

The T1G pin is one source for Timer1 gate control. It can be used to supply an external source to the Timer1 gate circuitry.

7.5.2.2 Timer0 Overflow Gate Operation

When Timer0 increments from FFh to 00h, a low-to-high pulse will automatically be generated and internally supplied to the Timer1 gate circuitry.

7.5.2.3 C1OUT/C2OUT Gate Operation

The outputs from the Comparator C1 and C2 modules can be used as gate sources for the Timer1 module.

7.5.3 TIMER1 GATE TOGGLE MODE

When Timer1 Gate Toggle mode is enabled, it is possible to measure the full-cycle length of a Timer1 gate signal, as opposed to the duration of a single level pulse.

The Timer1 gate source is routed through a flip-flop that changes state on every incrementing edge of the signal. See [Figure 7-4](#) for timing details.

Timer1 Gate Toggle mode is enabled by setting the T1GTM bit of the T1GCON register. When the T1GTM bit is cleared, the flip-flop is cleared and held clear. This is necessary in order to control which edge is measured.

| |
|---|
| Note: Enabling Toggle mode at the same time as changing the gate polarity may result in indeterminate operation. |
|---|

7.5.4 TIMER1 GATE SINGLE-PULSE MODE

When Timer1 Gate Single-Pulse mode is enabled, it is possible to capture a single pulse gate event. Timer1 Gate Single-Pulse mode is first enabled by setting the T1GSPM bit in the T1GCON register. Next, the T1GGO/ $\overline{\text{DONE}}$ bit in the T1GCON register must be set. The Timer1 will be fully enabled on the next incrementing edge. On the next trailing edge of the pulse, the T1GGO/ $\overline{\text{DONE}}$ bit will automatically be cleared. No other gate events will be allowed to increment Timer1 until the T1GGO/ $\overline{\text{DONE}}$ bit is once again set in software.

Clearing the T1GSPM bit of the T1GCON register will also clear the T1GGO/ $\overline{\text{DONE}}$ bit. See [Figure 7-5](#) for timing details.

Enabling the Toggle mode and the Single-Pulse mode simultaneously will permit both sections to work together. This allows the cycle times on the Timer1 gate source to be measured. See [Figure 7-6](#) for timing details.

7.5.5 TIMER1 GATE VALUE STATUS

When Timer1 gate value status is utilized, it is possible to read the most current level of the gate control value. The value is stored in the T1GVAL bit in the T1GCON register. The T1GVAL bit is valid even when the Timer1 gate is not enabled (TMR1GE bit is cleared).

7.5.6 TIMER1 GATE EVENT INTERRUPT

When Timer1 gate event interrupt is enabled, it is possible to generate an interrupt upon the completion of a gate event. When the falling edge of T1GVAL occurs, the TMR1GIF flag bit in the PIR1 register will be set. If the TMR1GIE bit in the PIE1 register is set, then an interrupt will be recognized.

The TMR1GIF flag bit operates even when the Timer1 gate is not enabled (TMR1GE bit is cleared).

7.6 Timer1 Interrupt

The Timer1 register pair (TMR1H:TMR1L) increments to FFFFh and rolls over to 0000h. When Timer1 rolls over, the Timer1 interrupt flag bit of the PIR1 register is set. To enable the interrupt on rollover, you must set these bits:

- TMR1ON bit of the T1CON register
- TMR1IE bit of the PIE1 register
- PEIE bit of the INTCON register
- GIE bit of the INTCON register

The interrupt is cleared by clearing the TMR1IF bit in the Interrupt Service Routine.

Note: The TMR1H:TMR1L register pair and the TMR1IF bit should be cleared before enabling interrupts.

7.7 Timer1 Operation During Sleep

Timer1 can only operate during Sleep when setup in Asynchronous Counter mode. In this mode, the clock source can be used to increment the counter. To set up the timer to wake the device:

- TMR1ON bit of the T1CON register must be set
- TMR1IE bit of the PIE1 register must be set
- PEIE bit of the INTCON register must be set
- T1SYNC bit of the T1CON register must be set
- TMR1CS bits of the T1CON register must be configured
- TMR1GE bit of the T1GCON register must be configured

The device will wake-up on an overflow and execute the next instructions. If the GIE bit of the INTCON register is set, the device will call the Interrupt Service Routine (0004h).

7.8 CCP Capture/Compare Time Base

The CCP module uses the TMR1H:TMR1L register pair as the time base when operating in Capture or Compare mode.

In Capture mode, the value in the TMR1H:TMR1L register pair is copied into the CCPR1H:CCPR1L register pair on a configured event.

In Compare mode, an event is triggered when the value CCPR1H:CCPR1L register pair matches the value in the TMR1H:TMR1L register pair. This event can be a Special Event Trigger.

For more information, see [Section 10.0 “Capture/Compare/PWM Modules”](#).

7.9 CCP Special Event Trigger

When the CCP is configured to trigger a special event, the trigger will clear the TMR1H:TMR1L register pair. This special event does not cause a Timer1 interrupt. The CCP module may still be configured to generate a CCP interrupt.

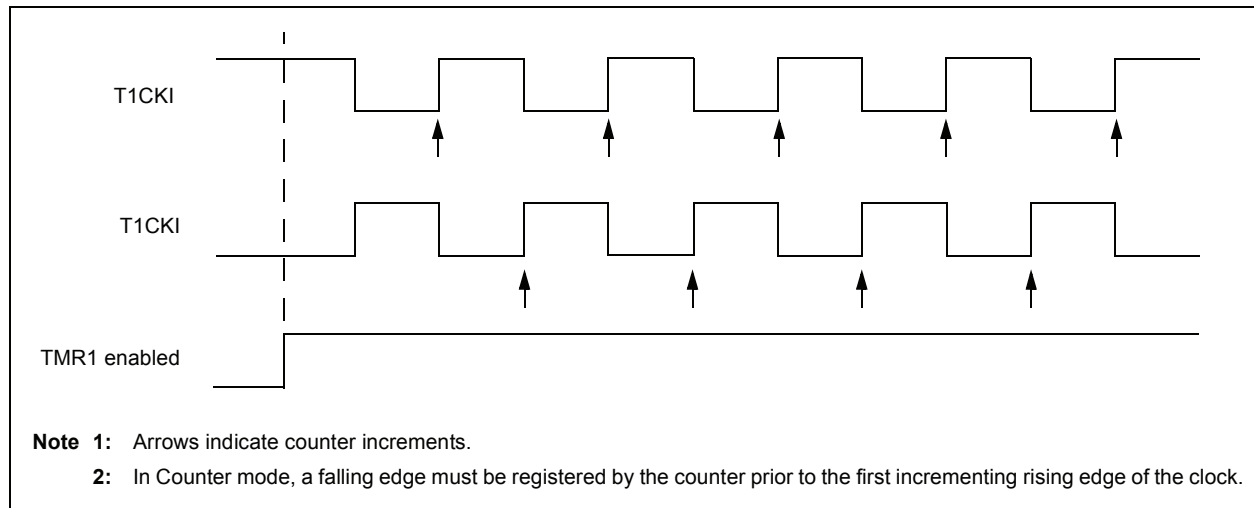
In this mode of operation, the CCPR1H:CCPR1L register pair becomes the period register for Timer1.

Timer1 should be synchronized to the FOSC/4 to utilize the Special Event Trigger. Asynchronous operation of Timer1 can cause a Special Event Trigger to be missed.

In the event that a write to TMR1H or TMR1L coincides with a Special Event Trigger from the CCP, the write will take precedence.

For more information, see [Section 12.2.5 “Special Event Trigger”](#).

FIGURE 7-2: TIMER1 INCREMENTING EDGE



PIC12F752/HV752

FIGURE 7-3: TIMER1 GATE COUNT ENABLE MODE

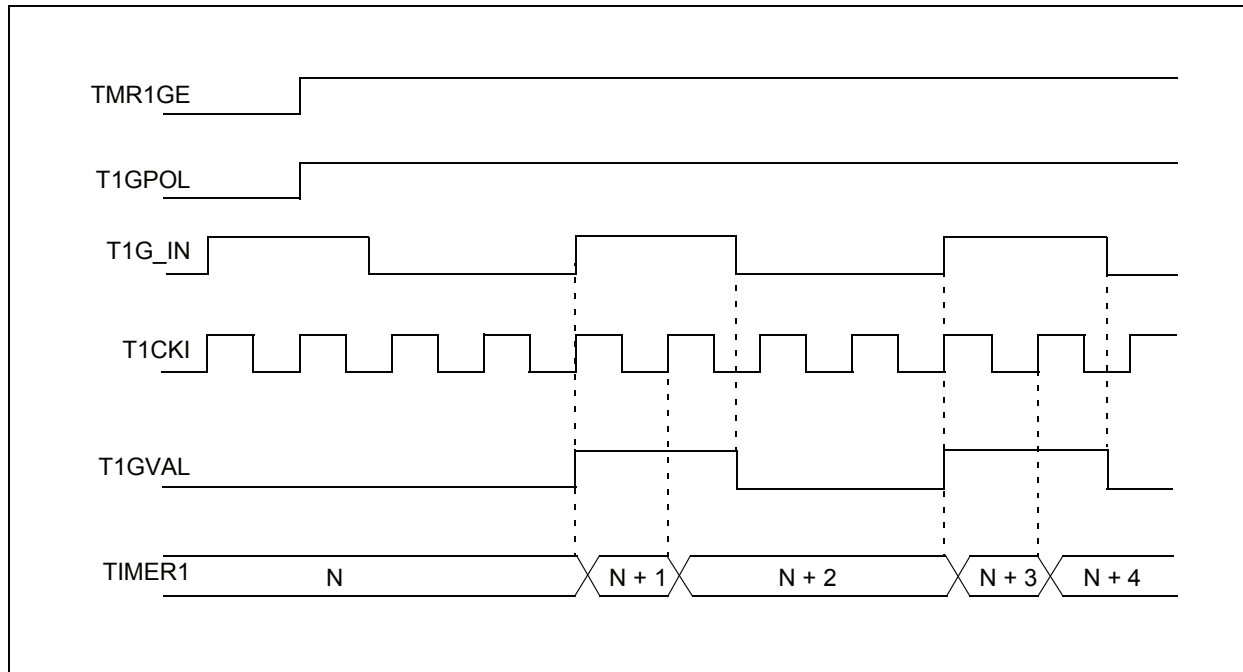


FIGURE 7-4: TIMER1 GATE TOGGLE MODE

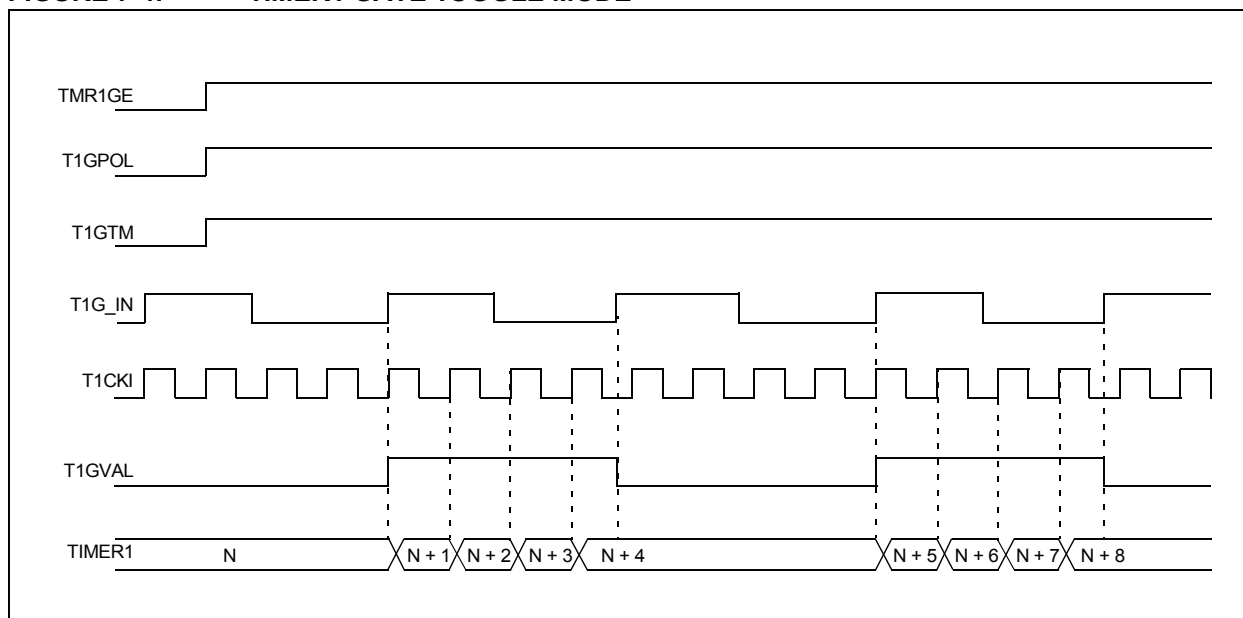
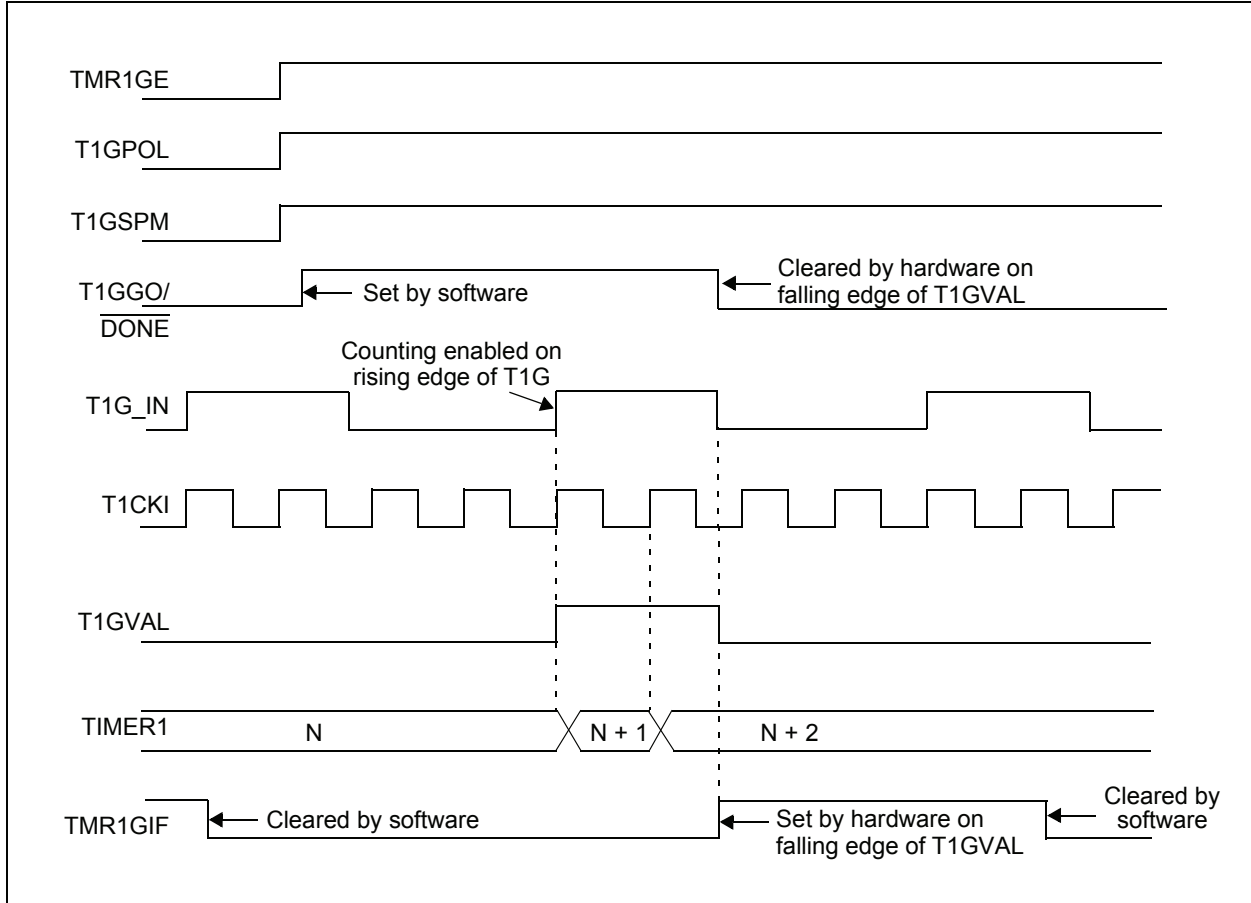
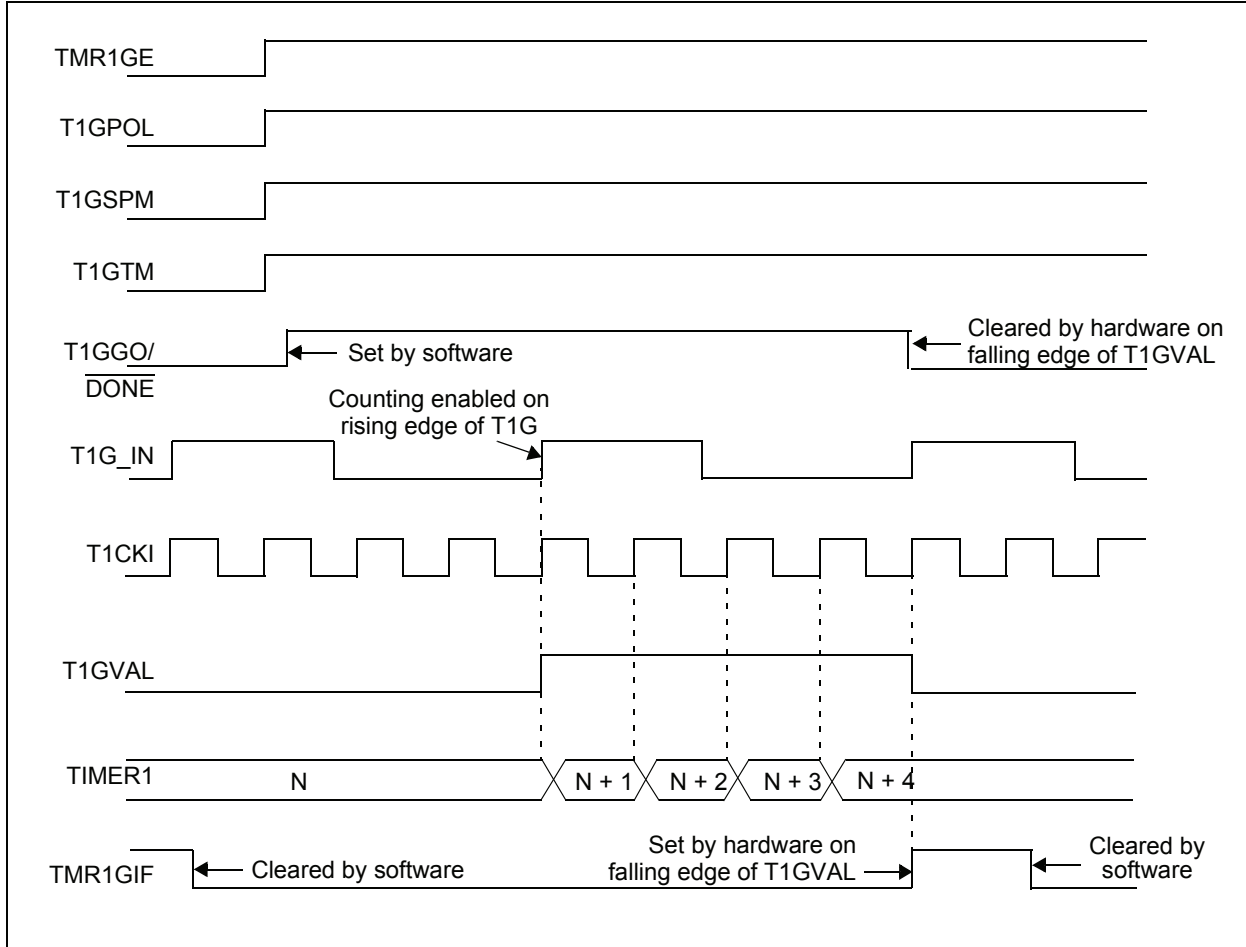


FIGURE 7-5: TIMER1 GATE SINGLE-PULSE MODE



PIC12F752/HV752

FIGURE 7-6: TIMER1 GATE SINGLE-PULSE AND TOGGLE COMBINED MODE



7.10 Timer1 Control Registers

REGISTER 7-1: T1CON: TIMER1 CONTROL REGISTER

| | | | | | | | |
|-------------|-------|-------------|-------|----------|--------|-------|--------|
| R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | U-0 | R/W-0 |
| TMR1CS<1:0> | | T1CKPS<1:0> | | Reserved | T1SYNC | — | TMR1ON |
| bit 7 | | | | | | bit 0 | |

Legend:

| | | |
|-------------------|------------------|------------------------------------|
| R = Readable bit | W = Writable bit | U = Unimplemented bit, read as '0' |
| -n = Value at POR | '1' = Bit is set | '0' = Bit is cleared |
| | | x = Bit is unknown |

- bit 7-6 **TMR1CS<1:0>**: Timer1 Clock Source Select bits
 11 = Temperature Sense Oscillator
 10 = External clock from T1CKI pin (on the rising edge)
 01 = Timer1 clock source is system clock (FOSC)
 00 = Timer1 clock source is instruction clock (FOSC/4)
- bit 5-4 **T1CKPS<1:0>**: Timer1 Input Clock Prescale Select bits
 11 = 1:8 Prescale value
 10 = 1:4 Prescale value
 01 = 1:2 Prescale value
 00 = 1:1 Prescale value
- bit 3 **Reserved**: Do not use.
- bit 2 **T1SYNC**: Timer1 External Clock Input Synchronization Control bit
TMR1CS<1:0> = 1X
 1 = Do not synchronize external clock input
 0 = Synchronize external clock input with system clock (Fosc)
- TMR1CS<1:0> = 0X
 This bit is ignored. Timer1 uses the internal clock when TMR1CS<1:0> = 1X.
- bit 1 **Unimplemented**: Read as '0'
- bit 0 **TMR1ON**: Timer1 On bit
 1 = Enables Timer1
 0 = Stops Timer1
 Clears Timer1 gate flip-flop

PIC12F752/HV752

REGISTER 7-2: T1GCON: TIMER1 GATE CONTROL REGISTER

| | | | | | | | |
|--------|--------|-------|--------|----------------|--------|------------|-------|
| R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R-x | R/W-0 | R/W-0 |
| TMR1GE | T1GPOL | T1GTM | T1GSPM | T1GGO/ DONE | T1GVAL | T1GSS<1:0> | |
| bit 7 | | | | | | bit 0 | |

Legend:

R = Readable bit

W = Writable bit

U = Unimplemented bit, read as '0'

-n = Value at POR

'1' = Bit is set

'0' = Bit is cleared

x = Bit is unknown

bit 7 **TMR1GE:** Timer1 Gate Enable bit

If TMR1ON = 0:

This bit is ignored

If TMR1ON = 1:

1 = Timer1 counting is controlled by the Timer1 gate function

0 = Timer1 counts regardless of Timer1 gate function

bit 6 **T1GPOL:** Timer1 Gate Polarity bit

1 = Timer1 gate is active-high (Timer1 counts when gate is high)

0 = Timer1 gate is active-low (Timer1 counts when gate is low)

bit 5 **T1GTM:** Timer1 Gate Toggle mode bit

1 = Timer1 Gate Toggle mode is enabled.

0 = Timer1 Gate Toggle mode is disabled and toggle flip-flop is cleared

Timer1 gate flip-flop toggles on every rising edge.

bit 4 **T1GSPM:** Timer1 Gate Single Pulse mode bit

1 = Timer1 Gate Single-Pulse mode is enabled and is controlling Timer1 gate

0 = Timer1 Gate Single-Pulse mode is disabled

bit 3 **T1GGO/DONE:** Timer1 Gate Single-Pulse Acquisition Status bit

1 = Timer1 gate single-pulse acquisition is ready, waiting for an edge

0 = Timer1 gate single-pulse acquisition has completed or has not been started

This bit is automatically cleared when T1GSPM is cleared.

bit 2 **T1GVAL:** Timer1 Gate Current State bit

Indicates the current state of the Timer1 gate that could be provided to TMR1H:TMR1L.

Unaffected by Timer1 Gate Enable (TMR1GE).

bit 1-0 **T1GSS<1:0>:** Timer1 Gate Source Select bits

11 = SYNCC2OUT

10 = SYNCC1OUT

01 = Timer0 overflow output

00 = Timer1 gate pin

TABLE 7-5: SUMMARY OF REGISTERS ASSOCIATED WITH TIMER1

| Name | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | Register on Page |
|---------|---|--------|-------------|--------|-------------------------|---------------------|------------|-----------|------------------|
| ANSELA | — | — | ANSA5 | ANSA4 | — | ANSA2 | ANSA1 | ANSA0 | 50 |
| APFCON | — | — | — | T1GSEL | — | COG1SEL | COG1O1SEL | COG1O0SEL | 46 |
| CCP1CON | — | — | DC1B<1:0> | | CCP1M<3:0> | | | | 83 |
| INTCON | GIE | PEIE | T0IE | INTE | IOCIE | T0IF | INTF | IOCIF | 20 |
| PIE1 | TMR1GIE | ADIE | — | — | — | HLTMR1IE | TMR2IE | TMR1IE | 21 |
| PIR1 | TMR1GIF | ADIF | — | — | — | HLTMR1IF | TMR2IF | TMR1IF | 23 |
| PORTA | — | — | RA5 | RA4 | RA3 | RA2 | RA1 | RA0 | 48 |
| TMR1H | Holding Register for the Most Significant Byte of the 16-bit TMR1 Register | | | | | | | | 59* |
| TMR1L | Holding Register for the Least Significant Byte of the 16-bit TMR1 Register | | | | | | | | 59* |
| TRISA | — | — | TRISA5 | TRISA4 | TRISA3 | TRISA2 | TRISA1 | TRISA0 | 48 |
| T1CON | TMR1CS<1:0> | | T1CKPS<1:0> | | Reserved | $\overline{T1SYNC}$ | — | TMR1ON | 67 |
| T1GCON | TMR1GE | T1GPOL | T1GTM | T1GSPM | $\overline{T1GGO/DONE}$ | $\overline{T1GVAL}$ | T1GSS<1:0> | | 68 |

Legend: x = unknown, u = unchanged, - = unimplemented, read as '0'. Shaded cells are not used by the Timer1 module.
 * Page provides register information.

PIC12F752/HV752

NOTES:

8.0 TIMER2 MODULE

The Timer2 module is an 8-bit timer with the following features:

- 8-bit timer register (TMR2)
- 8-bit period register (PR2)
- Interrupt on TMR2 match with PR2
- Software programmable prescaler (1:1, 1:4, 1:16)
- Software programmable postscaler (1:1 to 1:16)

See Figure 8-1 for a block diagram of Timer2.

8.1 Timer2 Operation

The clock input to the Timer2 module is the system instruction clock ($F_{osc}/4$). The clock is fed into the Timer2 prescaler, which has prescale options of 1:1, 1:4 or 1:16. The output of the prescaler is then used to increment the TMR2 register.

The values of TMR2 and PR2 are constantly compared to determine when they match. TMR2 will increment from 00h until it matches the value in PR2. When a match occurs, two things happen:

- TMR2 is reset to 00h on the next increment cycle.
- The Timer2 postscaler is incremented

The match output of the Timer2/PR2 comparator is then fed into the Timer2 postscaler. The postscaler has postscale options of 1:1 to 1:16 inclusive. The output of the Timer2 postscaler is used to set the TMR2IF interrupt flag bit in the PIR1 register.

The TMR2 and PR2 registers are both fully readable and writable. On any Reset, the TMR2 register is set to 00h and the PR2 register is set to FFh.

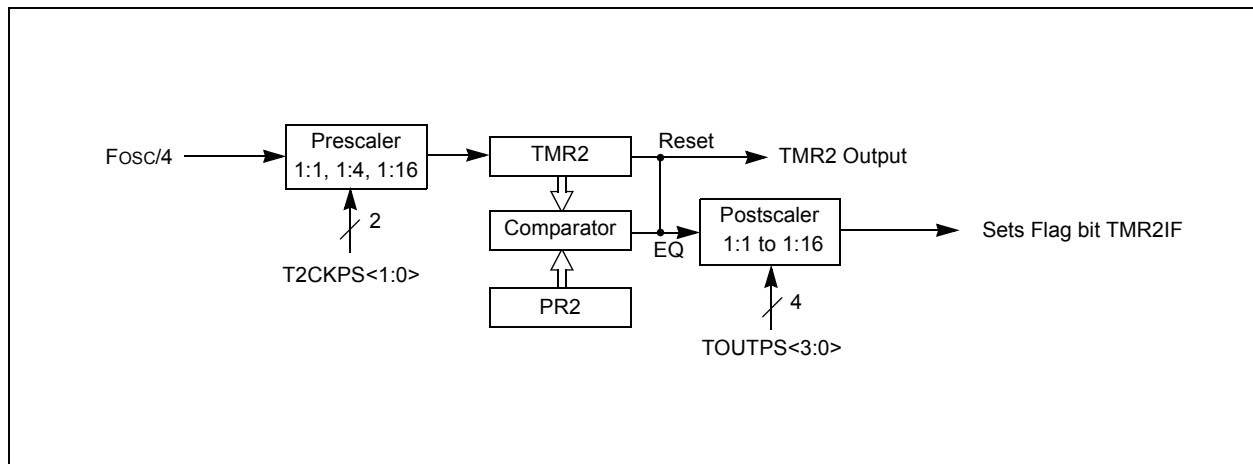
Timer2 is turned on by setting the TMR2ON bit in the T2CON register to a '1'. Timer2 is turned off by clearing the TMR2ON bit to a '0'.

The Timer2 prescaler is controlled by the T2CKPS bits in the T2CON register. The Timer2 postscaler is controlled by the TOUTPS bits in the T2CON register. The prescaler and postscaler counters are cleared when:

- A write to TMR2 occurs.
- A write to T2CON occurs.
- Any device Reset occurs (Power-on Reset, \overline{MCLR} Reset, Watchdog Timer Reset, or Brown-out Reset).

Note: TMR2 is not cleared when T2CON is written.

FIGURE 8-1: TIMER2 BLOCK DIAGRAM



PIC12F752/HV752

8.2 Timer2 Control Registers

REGISTER 8-1: T2CON: TIMER 2 CONTROL REGISTER

| | | | | | | | |
|-------|-------------|-------|-------|-------|--------|-------------|-------|
| U-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 |
| — | TOUTPS<3:0> | | | | TMR2ON | T2CKPS<1:0> | |
| bit 7 | | | | | | | bit 0 |

Legend:

R = Readable bit W = Writable bit U = Unimplemented bit, read as '0'
 -n = Value at POR '1' = Bit is set '0' = Bit is cleared x = Bit is unknown

- bit 7 **Unimplemented:** Read as '0'
- bit 6-3 **TOUTPS<3:0>:** Timer2 Output Postscaler Select bits
 - 0000 = 1:1 Postscaler
 - 0001 = 1:2 Postscaler
 - 0010 = 1:3 Postscaler
 - 0011 = 1:4 Postscaler
 - 0100 = 1:5 Postscaler
 - 0101 = 1:6 Postscaler
 - 0110 = 1:7 Postscaler
 - 0111 = 1:8 Postscaler
 - 1000 = 1:9 Postscaler
 - 1001 = 1:10 Postscaler
 - 1010 = 1:11 Postscaler
 - 1011 = 1:12 Postscaler
 - 1100 = 1:13 Postscaler
 - 1101 = 1:14 Postscaler
 - 1110 = 1:15 Postscaler
 - 1111 = 1:16 Postscaler
- bit 2 **TMR2ON:** Timer2 On bit
 - 1 = Timer2 is on
 - 0 = Timer2 is off
- bit 1-0 **T2CKPS<1:0>:** Timer2 Clock Prescale Select bits
 - 00 = Prescaler is 1
 - 01 = Prescaler is 4
 - 1x = Prescaler is 16

TABLE 8-1: SUMMARY OF REGISTERS ASSOCIATED WITH TIMER2

| Name | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | Register on Page |
|--------|--|-------------|-------|-------|--------|----------|-------------|--------|------------------|
| INTCON | GIE | PEIE | TOIE | INTE | IOCFIE | TOIF | INTF | IOCF | 20 |
| PIE1 | TMR1GIE | ADIE | — | — | — | HLTMR1IE | TMR2IE | TMR1IE | 21 |
| PIR1 | TMR1GIF | ADIF | — | — | — | HLTMR1IF | TMR2IF | TMR1IF | 23 |
| PR2 | Timer2 Module Period Register | | | | | | | | 71* |
| TMR2 | Holding Register for the 8-bit TMR2 Register | | | | | | | | 71* |
| T2CON | — | TOUTPS<3:0> | | | | TMR2ON | T2CKPS<1:0> | | 72 |

Legend: x = unknown, u = unchanged, - = unimplemented read as '0'. Shaded cells are not used for Timer2 module.
 * Page provides register information.

9.0 HARDWARE LIMIT TIMER (HLT) MODULE

The Hardware Limit Timer (HLT) module is a version of the Timer2-type modules. In addition to all the Timer2-type features, the HLT can be reset on rising and falling events from selected peripheral outputs.

The HLT primary purpose is to act as a timed hardware limit to be used in conjunction with asynchronous analog feedback applications. The external reset source synchronizes the HLTMR1 to an analog application.

In normal operation, the external reset source from the analog application should occur before the HLTMR1 matches the HLTMR1. This resets HLTMR1 for the next period and prevents the HLTMR1 Output from going active.

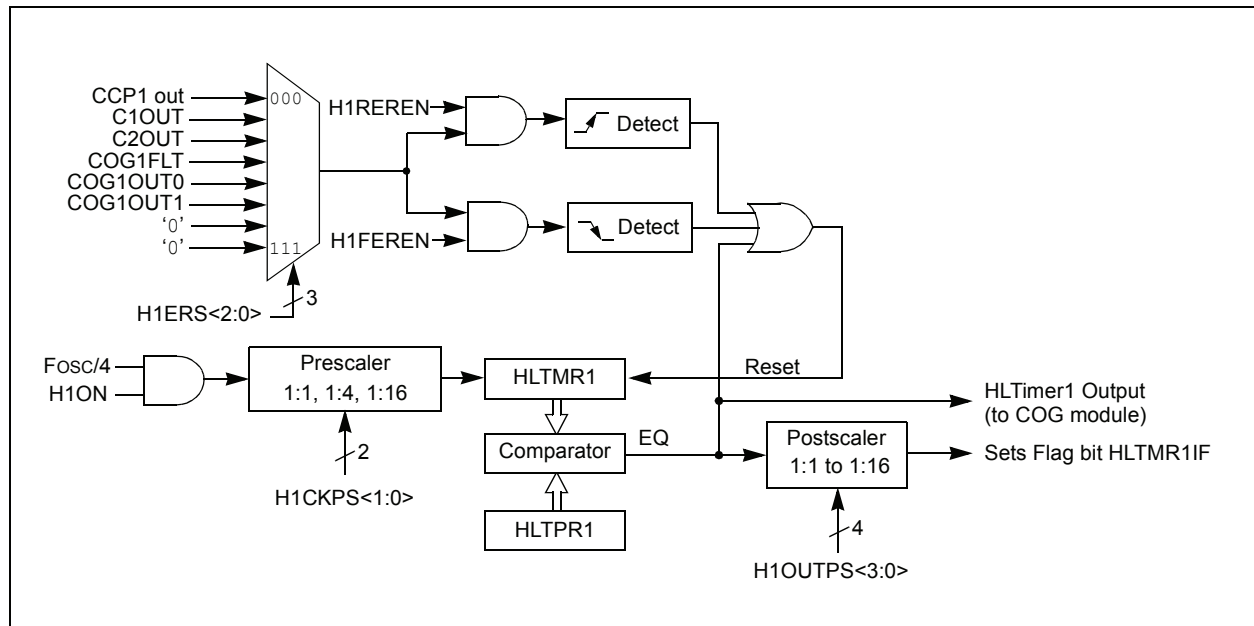
When the external reset source fails to generate a signal within the expected time, (allowing the HLTMR1 to match the HLTMR1), then the HLTMR1 Output becomes active.

The HLT module incorporates the following features:

- 8-bit Read-Write Timer Register (HLTMR1)
- 8-bit Read-Write Period register (HLTPR1)
- Software programmable prescaler:
 - 1:1
 - 1:4
 - 1:16
- Software programmable postscaler
 - 1:1 to 1:16, inclusive
- Interrupt on HLTMR1 match with HLTPR1
- 8 selectable timer Reset inputs (5 reserved)
- Reset on rising and falling event

Refer to [Figure 9-1](#) for a block diagram of the HLT.

FIGURE 9-1: HLTMR1 BLOCK DIAGRAM



PIC12F752/HV752

9.1 HLT Operation

The clock input to the HLT module is the system instruction clock (Fosc/4). HLTMR1 increments on each rising clock edge.

A 4-bit counter/prescaler on the clock input provides the following prescale options:

- Direct input
- Divide-by-4
- Divide-by-16

The prescale options are selected by the prescaler control bits, H1CKPS<1:0> of the HLT1CON0 register.

The value of HLTMR1 is compared to that of the Period register, HLTPR1, on each clock cycle. When the two values match, then the comparator generates a match signal as the HLTimer1 output. This signal also resets the value of HLTMR1 to 00h on the next clock rising edge and drives the output counter/postscaler (see [Section 9.2 “HLT Interrupt”](#)).

The HLTMR1 and HLTPR1 registers are both directly readable and writable. The HLTMR1 register is cleared on any device Reset, whereas the HLTPR1 register initializes to FFh. Both the prescaler and postscaler counters are cleared on any of the following events:

- A write to the HLTMR1 register
- A write to the HLT1CON0 register
- Power-on Reset (POR)
- Brown-out Reset (BOR)
- MCLR Reset
- Watchdog Timer (WDT) Reset
- Stack Overflow Reset
- Stack Underflow Reset
- RESET Instruction

Note: HLTMR1 is not cleared when HLT1CON0 is written.

9.2 HLT Interrupt

The HLT can also generate an optional device interrupt. The HLTMR1 output signal (HLTMR1-to-HLTPR1 match) provides the input for the 4-bit counter/postscaler. The overflow output of the postscaler sets the HLTMR1IF bit of the PIR1 register. The interrupt is enabled by setting the HLTMR1 Match Interrupt Enable bit, HLTMR1IE of the PIE1 register.

A range of 16 postscale options (from 1:1 through 1:16 inclusive) can be selected with the postscaler control bits, H1OUTPS<3:0>, of the HLT1CON0 register.

9.3 Peripheral Resets

Resets driven from the selected peripheral output prevents the HLTMR1 from matching the HLTPR1 register and generating an output. In this manner, the HLT can be used as a hardware time limit to other peripherals.

In this device, the primary purpose of the HLT is to limit the COG PWM duty cycle. Normally, the COG operation uses analog feedback to determine the PWM duty cycle. The same feedback signal is used as an HLT Reset input. The HLTPR1 register is set to occur at the maximum allowed duty cycle. If the analog feedback to the COG exceeds the maximum time, then an HLTMR1-to-HLTPR1 match will occur and generate the output needed to limit the COG drive output.

The HLTMR1 can be reset by one of several selectable peripheral sources. Reset inputs include:

- CCP1 output
- Comparator 1 output
- Comparator 2 output

The Reset input is selected with the H1ERS<2:0> bits of the HLT1CON1 register.

HLTMR1 Resets are synchronous with the HLT clock. In other words, HLTMR1 is cleared on the rising edge of the HLT clock after the enabled Reset event occurs.

The Reset can be enabled to occur on the rising and falling input event. Rising and falling event enables are selected with the respective H1REREN and H1FEREN bits of the HLT1CON1 register. External Resets do not cause an HLTMR1 output event.

9.4 HLTimer1 Output

The unscaled output of HLTMR1 is available only to the COG module, where it is used as a selectable limit to the maximum COG period.

9.5 HLT Operation During Sleep

The HLT cannot be operated while the processor is in Sleep mode. The contents of the HLTMR1 register will remain unchanged while the processor is in Sleep mode.

9.6 HLT Control Registers

REGISTER 9-1: HLT1CON0: HLT1 CONTROL REGISTER 0

| | | | | | | | |
|-------|--------------|---------|---------|---------|-------------|---------|---------|
| U-0 | R/W-0/0 | R/W-0/0 | R/W-0/0 | R/W-0/0 | R/W-0/0 | R/W-0/0 | R/W-0/0 |
| — | H1OUTPS<3:0> | | | H1ON | H1CKPS<1:0> | | |
| bit 7 | | | | | | | bit 0 |

Legend:

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

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

bit 6-3 **H1OUTPS<3:0>:** Hardware Limit Timer 1 Output Postscaler Select bits

0000 = 1:1 Postscaler
 0001 = 1:2 Postscaler
 0010 = 1:3 Postscaler
 0011 = 1:4 Postscaler
 0100 = 1:5 Postscaler
 0101 = 1:6 Postscaler
 0110 = 1:7 Postscaler
 0111 = 1:8 Postscaler
 1000 = 1:9 Postscaler
 1001 = 1:10 Postscaler
 1010 = 1:11 Postscaler
 1011 = 1:12 Postscaler
 1100 = 1:13 Postscaler
 1101 = 1:14 Postscaler
 1110 = 1:15 Postscaler
 1111 = 1:16 Postscaler

bit 2 **H1ON:** Hardware Limit Timer 1 On bit

1 = Timer is on
 0 = Timer is off

bit 1-0 **H1CKPS<1:0>:** Hardware Limit Timer 1 Clock Prescale Select bits

00 = Prescaler is 1
 01 = Prescaler is 4
 1x = Prescaler is 16

PIC12F752/HV752

REGISTER 9-2: HLT1CON1: HLT1 CONTROL REGISTER 1

| | | | | | | | |
|-------|-----|-----|------------|---------|---------|---------|---------|
| U-0 | U-0 | U-0 | R/W-0/0 | R/W-0/0 | R/W-0/0 | R/W-0/0 | R/W-0/0 |
| — | — | — | H1ERS<2:0> | | | H1FEREN | H1REREN |
| bit 7 | | | | | | | bit 0 |

Legend:

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

- bit 7-5 **Unimplemented:** Read as '0'
- bit 4-2 **H1ERS<2:0>:** Hardware Limit Timer 1 Peripheral Reset Select bits
 - 000 = CCP1 Out
 - 001 = C1OUT
 - 010 = C2OUT
 - 011 = COG1FLT
 - 100 = COG1OUT0
 - 101 = COG1OUT1
 - 110 = Reserved - '0' input
 - 111 = Reserved - '0' input
- bit 1 **H1FEREN:** Hardware Limit Timer 1 Falling Edge Reset Enable bit
 - 1 = HLTMR1 will reset on the first clock after a falling edge of selected Reset source
 - 0 = Falling edges of selected source have no effect
- bit 0 **H1REREN:** Hardware Limit Timer 1 Rising Edge Reset Enable bit
 - 1 = HLTMR1 will reset on the first clock after a rising edge of selected Reset source
 - 0 = Rising edges of selected source have no effect

TABLE 9-1: SUMMARY OF REGISTERS ASSOCIATED WITH HLT

| Name | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | Register on Page |
|----------|---|--------------|------------|------------|------------|-------------|---------|---------|------------------|
| CCP1CON | — | — | DC1B<1:0> | | CCP1M<3:0> | | | | 83 |
| CM1CON0 | C1ON | C1OUT | C1OE | C1POL | C1ZLF | C1SP | C1HYS | C1SYNC | 127 |
| CM1CON1 | C1INTP | C1INTN | C1PCH<1:0> | | — | — | — | C1NCH0 | 128 |
| CM2CON0 | C2ON | C2OUT | C2OE | C2POL | C2ZLF | C2SP | C2HYS | C2SYNC | 127 |
| CM2CON1 | C2INTP | C2INTN | C2PCH<1:0> | | — | — | — | C2NCH0 | 128 |
| INTCON | GIE | PEIE | T0IE | INTE | IOCIE | T0IF | INTF | IOCIF | 20 |
| PIE1 | TMR1GIE | ADIE | — | — | — | HLTMR1IE | TMR2IE | TMR1IE | 21 |
| PIR1 | TMR1GIF | ADIF | — | — | — | HLTMR1IF | TMR2IF | TMR1IF | 23 |
| HLTMR1 | Holding Register for the 8-bit Hardware Limit Timer1 Register | | | | | | | | 73* |
| HLTPR1 | HLTMR1 Module Period Register | | | | | | | | 73* |
| HLT1CON0 | — | H1OUTPS<3:0> | | | H1ON | H1CKPS<1:0> | | | 75 |
| HLT1CON1 | — | — | — | H1ERS<2:0> | | | H1FEREN | H1REREN | 76 |

Legend: — = unimplemented location, read as '0'. Shaded cells do not affect the HLT module operation.
* Page provides register information.

10.0 CAPTURE/COMPARE/PWM MODULES

The Capture/Compare/PWM module is a peripheral which allows the user to time and control different events, and to generate Pulse-Width Modulation (PWM) signals. In Capture mode, the peripheral allows the timing of the duration of an event. The Compare mode allows the user to trigger an external event when a predetermined amount of time has expired. The PWM mode can generate Pulse-Width Modulated signals of varying frequency and duty cycle.

10.1 Capture Mode

Capture mode makes use of the 16-bit Timer1 resource. When an event occurs on the CCP1 pin, the 16-bit CCPR1H:CCPR1L register pair captures and stores the 16-bit value of the TMR1H:TMR1L register pair, respectively. An event is defined as one of the following and is configured by the CCP1M<3:0> bits of the CCP1CON register:

- Every falling edge
- Every rising edge
- Every 4th rising edge
- Every 16th rising edge

When a capture is made, the Interrupt Request Flag bit CCP1IF of the PIR2 register is set. The interrupt flag must be cleared in software. If another capture occurs before the value in the CCPR1H, CCPR1L register pair is read, the old captured value is overwritten by the new captured value.

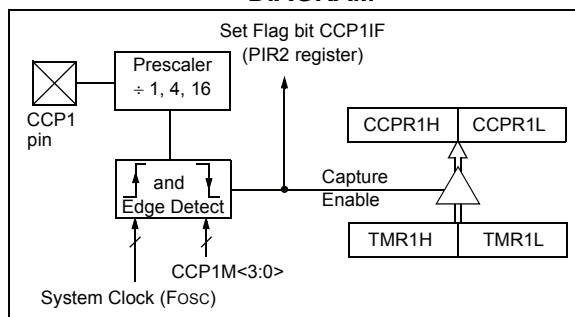
Figure 10-1 shows a simplified diagram of the Capture operation.

10.1.1 CCP1 PIN CONFIGURATION

In Capture mode, the CCP1 pin should be configured as an input by setting the associated TRIS control bit.

Note: If the CCP1 pin is configured as an output, a write to the port can cause a capture condition.

FIGURE 10-1: CAPTURE MODE OPERATION BLOCK DIAGRAM



10.1.2 TIMER1 MODE RESOURCE

Timer1 must be running in Timer mode or Synchronized Counter mode for the CCP1 module to use the capture feature. In Asynchronous Counter mode, the capture operation may not work.

See Section 7.0 “Timer1 Module with Gate Control” for more information on configuring Timer1.

10.1.3 SOFTWARE INTERRUPT MODE

When the Capture mode is changed, a false capture interrupt may be generated. The user should keep the CCP1IE interrupt enable bit of the PIE2 register clear to avoid false interrupts. Additionally, the user should clear the CCP1IF interrupt flag bit of the PIR2 register following any change in Operating mode.

Note: Clocking Timer1 from the system clock (Fosc) should not be used in Capture mode. In order for Capture mode to recognize the trigger event on the CCP1 pin, Timer1 must be clocked from the instruction clock (Fosc/4) or from an external clock source.

10.1.4 CCP1 PRESCALER

There are four prescaler settings specified by the CCP1M<3:0> bits of the CCP1CON register. Whenever the CCP1 module is turned off, or the CCP1 module is not in Capture mode, the prescaler counter is cleared. Any Reset will clear the prescaler counter.

Switching from one capture prescaler to another does not clear the prescaler and may generate a false interrupt. To avoid this unexpected operation, turn the module off by clearing the CCP1CON register before changing the prescaler. Example 10-1 demonstrates the code to perform this function.

EXAMPLE 10-1: CHANGING BETWEEN CAPTURE PRESCALERS

```
BANKSEL CCP1CON    ;Set Bank bits to point
                   ;to CCP1CON
CLRWF  CCP1CON     ;Turn CCP1 module off
MOVLW  NEW_CAPT_PS;Load the W reg with
                   ;the new prescaler
MOVWF  CCP1CON     ;move value and CCP1 ON
MOVLW  CCP1CON     ;Load CCP1CON with this
                   ;value
```

PIC12F752/HV752

10.1.5 CAPTURE DURING SLEEP

Capture mode depends upon the Timer1 module for proper operation. If the Timer1 clock input source is a clock that is not disabled during Sleep, Timer1 will continue to operate and Capture mode will operate during Sleep to wake the device. The T1CKI is an example of a clock source that will operate during Sleep.

When the input source to Timer1 is disabled during Sleep, such as the HFINTOSC, Timer1 will not increment during Sleep. When the device wakes from Sleep, Timer1 will continue from its previous state.

TABLE 10-1: SUMMARY OF REGISTERS ASSOCIATED WITH CAPTURE

| Name | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | Register on Page |
|---------|---|--------|-------------|--------|-----------------------|---------------------|------------|--------|------------------|
| CCP1CON | — | — | DC1B<1:0> | | CCP1M<3:0> | | | | 83 |
| CCPR1L | Capture/Compare/PWM Register x Low Byte (LSB) | | | | | | | | 77 |
| CCPR1H | Capture/Compare/PWM Register x High Byte (MSB) | | | | | | | | 77 |
| INTCON | GIE | PEIE | T0IE | INTE | IOCFE | T0IF | INTF | IOCF | 20 |
| PIE1 | TMR1GIE | ADIE | — | — | — | HLTMR1IE | TMR2IE | TMR1IE | 21 |
| PIE2 | — | — | C2IE | C1IE | — | COG1IE | — | CCP1IE | 22 |
| PIR1 | TMR1GIF | ADIF | — | — | — | HLTMR1IF | TMR2IF | TMR1IF | 23 |
| PIR2 | — | — | C2IF | C1IF | — | COG1IF | — | CCP1IF | 24 |
| T1CON | TMR1CS<1:0> | | T1CKPS<1:0> | | Reserved | $\overline{T1SYNC}$ | — | TMR1ON | 67 |
| T1GCON | TMR1GE | T1GPOL | T1GTM | T1GSPM | T1GGO/ DONE | $\overline{T1GVAL}$ | T1GSS<1:0> | | 68 |
| TMR1H | Holding Register for the Most Significant Byte of the 16-bit TMR1 Register | | | | | | | | 59* |
| TMR1L | Holding Register for the Least Significant Byte of the 16-bit TMR1 Register | | | | | | | | 59* |
| TRISA | — | — | TRISA5 | TRISA4 | TRISA3 ⁽¹⁾ | TRISA2 | TRISA1 | TRISA0 | 48 |

Legend: — = Unimplemented location, read as '0'. Shaded cells are not used by Capture mode.

* Page provides register information.

Note 1: TRISA3 always reads '1'.

10.2 Compare Mode

Compare mode makes use of the 16-bit Timer1 resource. The 16-bit value of the CCPR1H:CCPR1L register pair is constantly compared against the 16-bit value of the TMR1H:TMR1L register pair. When a match occurs, one of the following events can occur:

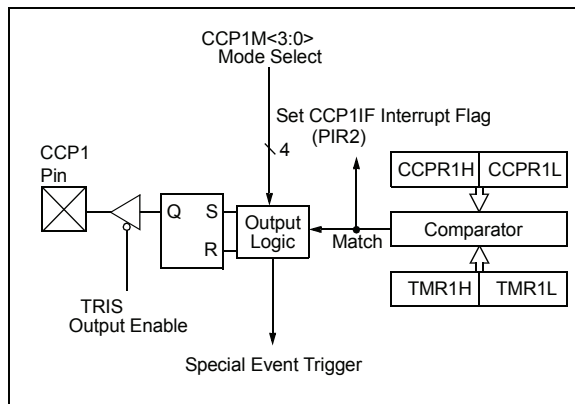
- Toggle the CCP1 output
- Set the CCP1 output
- Clear the CCP1 output
- Generate a Special Event Trigger
- Generate a Software Interrupt

The action on the pin is based on the value of the CCP1M<3:0> control bits of the CCP1CON register. At the same time, the interrupt flag CCP1IF bit is set.

All Compare modes can generate an interrupt.

Figure 10-2 shows a simplified diagram of the Compare operation.

FIGURE 10-2: COMPARE MODE OPERATION BLOCK DIAGRAM



10.2.1 CCP1 PIN CONFIGURATION

The user must configure the CCP1 pin as an output by clearing the associated TRIS bit.

Note: Clearing the CCP1CON register will force the CCP1 compare output latch to the default low level. This is not the PORT I/O data latch.

10.2.2 TIMER1 MODE RESOURCE

In Compare mode, Timer1 must be running in either Timer mode or Synchronized Counter mode. The compare operation may not work in Asynchronous Counter mode.

See Section 7.0 “Timer1 Module with Gate Control” for more information on configuring Timer1.

Note: Clocking Timer1 from the system clock (FOSC) should not be used in Compare mode. In order for Compare mode to recognize the trigger event on the CCP1 pin, Timer1 must be clocked from the instruction clock (FOSC/4) or from an external clock source.

10.2.3 SOFTWARE INTERRUPT MODE

When Generate Software Interrupt mode is chosen (CCP1M<3:0> = 1010), the CCP1 module does not assert control of the CCP1 pin (see the CCP1CON register).

10.2.4 SPECIAL EVENT TRIGGER

When Special Event Trigger mode is chosen (CCP1M<3:0> = 1011), the CCP1 module does the following:

- Resets Timer1
- Starts an ADC conversion if ADC is enabled

The CCP1 module does not assert control of the CCP1 pin in this mode.

The Special Event Trigger output of the CCP1 occurs immediately upon a match between the TMR1H, TMR1L register pair and the CCPR1H, CCPR1L register pair. The TMR1H, TMR1L register pair is not reset until the next rising edge of the Timer1 clock. The Special Event Trigger output starts an A/D conversion (if the A/D module is enabled). This allows the CCPR1H, CCPR1L register pair to effectively provide a 16-bit programmable period register for Timer1.

TABLE 10-2: SPECIAL EVENT TRIGGER

| Device | CCP1 |
|------------|------|
| PIC12F752 | CCP1 |
| PIC12HV752 | |

Refer to Section 12.0 “Analog-to-Digital Converter (ADC) Module” for more information.

- Note 1:** The Special Event Trigger from the CCP module does not set interrupt flag bit TMR1IF of the PIR1 register.
- 2:** Removing the match condition by changing the contents of the CCPR1H and CCPR1L register pair, between the clock edge that generates the Special Event Trigger and the clock edge that generates the Timer1 Reset, will preclude the Reset from occurring.

PIC12F752/HV752

10.2.5 COMPARE DURING SLEEP

The Compare mode is dependent upon the system clock (Fosc) for proper operation. Since Fosc is shut down during Sleep mode, the Compare mode will not function properly during Sleep.

TABLE 10-3: SUMMARY OF REGISTERS ASSOCIATED WITH COMPARE

| Name | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | Register on Page |
|---------|---|--------|-------------|--------|-----------------------|----------|------------|--------|------------------|
| CCP1CON | — | — | DC1B<1:0> | | CCP1M<3:0> | | | | 83 |
| CCPR1L | Capture/Compare/PWM Register 1 Low Byte (LSB) | | | | | | | | 77 |
| CCPR1H | Capture/Compare/PWM Register 1 High Byte (MSB) | | | | | | | | 77 |
| INTCON | GIE | PEIE | T0IE | INTE | IOCIE | T0IF | INTF | IOCIF | 20 |
| PIE1 | TMR1GIE | ADIE | — | — | — | HLTMR1IE | TMR2IE | TMR1IE | 20 |
| PIE2 | — | — | C2IE | C1IE | — | COG1IE | — | CCP1IE | 20 |
| PIR1 | TMR1GIF | ADIF | — | — | — | HLTMR1IF | TMR2IF | TMR1IF | 20 |
| PIR2 | — | — | C2IF | C1IF | — | COG1IF | — | CCP1IF | 20 |
| T1CON | TMR1CS<1:0> | | T1CKPS<1:0> | | Reserved | T1SYNC | — | TMR1ON | 67 |
| T1GCON | TMR1GE | T1GPOL | T1GTM | T1GSPM | T1GGO/DONE | T1GVAL | T1GSS<1:0> | | 68 |
| TMR1H | Holding Register for the Most Significant Byte of the 16-bit TMR1 Register | | | | | | | | 59* |
| TMR1L | Holding Register for the Least Significant Byte of the 16-bit TMR1 Register | | | | | | | | 59* |
| TRISA | — | — | TRISA5 | TRISA4 | TRISA3 ⁽¹⁾ | TRISA2 | TRISA1 | TRISA0 | 48 |

Legend: — = Unimplemented location, read as '0'. Shaded cells are not used by Compare mode.

* Page provides register information.

Note 1: TRISA3 always reads '1'.

10.3 PWM Overview

Pulse-Width Modulation (PWM) is a scheme that provides power to a load by switching quickly between fully on and fully off states. The PWM signal resembles a square wave where the high portion of the signal is considered the on state and the low portion of the signal is considered the off state. The high portion, also known as the pulse width, can vary in time and is defined in steps. A larger number of steps applied, which lengthens the pulse width, also supplies more power to the load. Lowering the number of steps applied, which shortens the pulse width, supplies less power. The PWM period is defined as the duration of one complete cycle or the total amount of on and off time combined.

PWM resolution defines the maximum number of steps that can be present in a single PWM period. A higher resolution allows for more precise control of the pulse width time and in turn the power that is applied to the load.

The term duty cycle describes the proportion of the on time to the off time and is expressed in percentages, where 0% is fully off and 100% is fully on. A lower duty cycle corresponds to less power applied and a higher duty cycle corresponds to more power applied.

Figure 10-3 shows a typical waveform of the PWM signal.

10.3.1 STANDARD PWM OPERATION

The standard PWM mode generates a Pulse-Width modulation (PWM) signal on the CCP1 pin with up to 10 bits of resolution. The period, duty cycle, and resolution are controlled by the following registers:

- PR2 registers
- T2CON registers
- CCPR1L registers
- CCP1CON registers

Figure 10-4 shows a simplified block diagram of PWM operation.

Note 1: The corresponding TRIS bit must be cleared to enable the PWM output on the CCP1 pin.

2: Clearing the CCP1CON register will relinquish control of the CCP1 pin.

FIGURE 10-3: CCP1 PWM OUTPUT SIGNAL

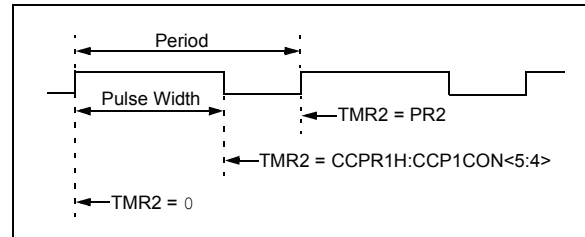
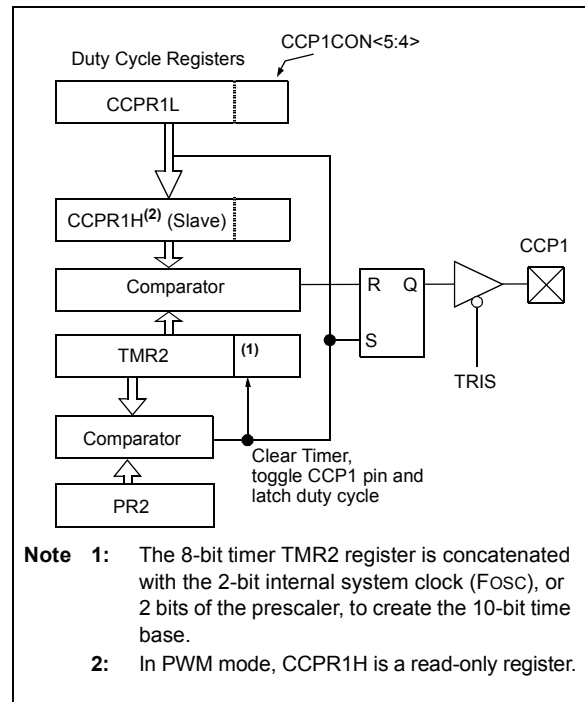


FIGURE 10-4: SIMPLIFIED PWM BLOCK DIAGRAM



PIC12F752/HV752

10.3.2 SETUP FOR PWM OPERATION

The following steps should be taken when configuring the CCP1 module for standard PWM operation:

1. Disable the CCP1 pin output driver by setting the associated TRIS bit.
2. Load the PR2 register with the PWM period value.
3. Configure the CCP1 module for the PWM mode by loading the CCP1CON register with the appropriate values.
4. Load the CCPR1L register and the DC1B<1:0> bits of the CCP1CON register, with the PWM duty cycle value.
5. Configure and start Timer2:
 - Clear the TMR2IF interrupt flag bit of the PIR1 register. See Note below.
 - Configure the T2CKPS bits of the T2CON register with the Timer prescale value.
 - Enable the Timer by setting the TMR2ON bit of the T2CON register.
6. Enable PWM output pin:
 - Wait until the Timer overflows and the TMR2IF bit of the PIR1 register is set. See Note below.
 - Enable the CCP1 pin output driver by clearing the associated TRIS bit.

Note: In order to send a complete duty cycle and period on the first PWM output, the above steps must be included in the setup sequence. If it is not critical to start with a complete PWM signal on the first output, then step 6 may be ignored.

10.3.3 PWM PERIOD

The PWM period is specified by the PR2 register of Timer2. The PWM period can be calculated using the formula of Equation 10-1.

EQUATION 10-1: PWM PERIOD

$$PWM\ Period = [(PR2) + 1] \cdot 4 \cdot TOSC \cdot (TMR2\ Prescale\ Value)$$

Note 1: $TOSC = 1/FOSC$

When TMR2 is equal to PR2, the following three events occur on the next increment cycle:

- TMR2 is cleared
- The CCP1 pin is set. (Exception: If the PWM duty cycle = 0%, the pin will not be set.)
- The PWM duty cycle is latched from CCPR1L into CCPR1H.

Note: The Timer postscaler (see Section 8.1 “Timer2 Operation”) is not used in the determination of the PWM frequency.

10.3.4 PWM DUTY CYCLE

The PWM duty cycle is specified by writing a 10-bit value to multiple registers: CCPR1L register and DC1B<1:0> bits of the CCP1CON register. The CCPR1L contains the eight MSBs and the DC1B<1:0> bits of the CCP1CON register contain the two LSBs. CCPR1L and DC1B<1:0> bits of the CCP1CON register can be written to at any time. The duty cycle value is not latched into CCPR1H until after the period completes (i.e., a match between PR2 and TMR2 registers occurs). While using the PWM, the CCPR1H register is read-only.

Equation 10-2 is used to calculate the PWM pulse width.

Equation 10-3 is used to calculate the PWM duty cycle ratio.

EQUATION 10-2: PULSE WIDTH

$$Pulse\ Width = (CCPR1L:CCP1CON<5:4>) \cdot TOSC \cdot (TMR2\ Prescale\ Value)$$

EQUATION 10-3: DUTY CYCLE RATIO

$$Duty\ Cycle\ Ratio = \frac{(CCPRxL:CCPxCON<5:4>)}{4(PR_x + 1)}$$

The CCPR1H register and a 2-bit internal latch are used to double buffer the PWM duty cycle. This double buffering is essential for glitchless PWM operation.

The 8-bit timer TMR2 register is concatenated with either the 2-bit internal system clock (FOSC), or 2 bits of the prescaler, to create the 10-bit time base. The system clock is used if the Timer2 prescaler is set to 1:1.

When the 10-bit time base matches the CCPR1H and 2-bit latch, then the CCP1 pin is cleared (see Figure 10-4).

10.4 CCP Control Registers

REGISTER 10-1: CCP1CON: CCP1 CONTROL REGISTER

| U-0 | U-0 | R/W-0/0 | R/W-0/0 | R/W-0/0 | R/W-0/0 | R/W-0/0 | R/W-0/0 | |
|-------|-----|-----------|---------|------------|---------|---------|---------|-------|
| — | — | DC1B<1:0> | | CCP1M<3:0> | | | | |
| bit 7 | | | | | | | | bit 0 |

Legend:

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

bit 7-6 **Unimplemented:** Read as '0'

bit 5-4 **DC1B<1:0>:** PWM Duty Cycle Least Significant bits

Capture mode:

Unused

Compare mode:

Unused

PWM mode:

These bits are the two LSBs of the PWM duty cycle. The eight MSBs are found in CCP1L.

bit 3-0 **CCP1M<3:0>:** CCP1 Mode Select bits

0000 = Capture/Compare/PWM off (resets CCP1 module)

0001 = Reserved

0010 = Compare mode: toggle output on match

0011 = Reserved

0100 = Capture mode: every falling edge

0101 = Capture mode: every rising edge

0110 = Capture mode: every 4th rising edge

0111 = Capture mode: every 16th rising edge

1000 = Compare mode: initialize CCP1 pin low; set output on compare match (set CCP1IF)

1001 = Compare mode: initialize CCP1 pin high; clear output on compare match (set CCP1IF)

1010 = Compare mode: generate software interrupt only; CCP1 pin reverts to I/O state

1011 = Compare mode: Special Event Trigger (CCP1 resets Timer, sets CCP1IF bit, and starts A/D conversion if A/D module is enabled)

11xx = PWM mode

PIC12F752/HV752

NOTES:

11.0 COMPLEMENTARY OUTPUT GENERATOR (COG) MODULE

The primary purpose of the Complementary Output Generator (COG) is to convert a single output PWM signal into a two output complementary PWM signal. The COG can also convert two separate input events into a single or complementary PWM output.

The COG PWM frequency and duty cycle are determined by a rising event input and a falling event input. The rising event and falling event may be the same source. Sources may be synchronous or asynchronous to the COG_clock.

The rate at which the rising event occurs determines the PWM frequency. The time from the rising event input to the falling event input determines the duty cycle.

A selectable clock input is used to generate the phase delay, blanking and dead-band times.

A simplified block diagram of the COG is shown in [Figure 11-1](#).

The COG module has the following features:

- Selectable clock source
- Selectable rising event source
- Selectable falling event source
- Selectable edge or level event sensitivity
- Independent output enables
- Independent output polarity selection
- Phase delay
- Dead-band control with independent rising and falling event dead-band times
- Blanking control with independent rising and falling event blanking times
- Auto-shutdown control with:
 - Selectable shutdown sources
 - Auto-restart enable
 - Auto-shutdown pin override control

11.1 Fundamental Operation

The COG generates a two output complementary PWM waveform from rising and falling event sources. In the simplest configuration, the rising and falling event sources are the same signal, which is a PWM signal with the desired period and duty cycle. The COG converts this single PWM input into a dual complementary PWM output. The frequency and duty cycle of the dual PWM output match those of the single input PWM signal. The off-to-on transition of each output can be delayed from the on-to-off transition of the other output, thereby, creating a time immediately after the PWM transition where neither output is driven. This is referred to as dead time and is covered in [Section 11.5 “Dead-Band Control”](#).

A typical operating waveform, with dead band, generated from a single CCP1 input is signal is shown in [Figure 11-2](#).

The COG can also generate a PWM waveform from a periodic rising event and a separate falling event. In this case, the falling event is usually derived from analog feedback within the external PWM driver circuit. In this configuration, high power switching transients may trigger a false falling event that needs to be blanked out. The COG can be configured to blank falling (and rising) event inputs for a period of time immediately following the rising (and falling) event drive output. This is referred to as input blanking and is covered in [Section 11.6 “Blanking Control”](#).

It may be necessary to guard against the possibility of circuit faults. In this case, the active drive must be terminated before the Fault condition causes damage. This is referred to as auto-shutdown and is covered in [Section 11.8 “Auto-shutdown Control”](#).

A feedback falling event arriving too late or not at all can be terminated with auto-shutdown or by using one of the event inputs that is logically OR'd with the hardware limit timer (HLT). See [Section 9.0 “Hardware Limit Timer \(HLT\) Module”](#) for more information about the HLT.

The COG can be configured to operate in phase delayed conjunction with another PWM. The active drive cycle is delayed from the rising event by a phase delay timer. Phase delay is covered in more detail in [Section 11.7 “Phase Delay”](#).

A typical operating waveform, with phase delay and dead band, generated from a single CCP1 input is shown in [Figure 11-3](#).

FIGURE 11-1: SIMPLIFIED COG BLOCK DIAGRAM

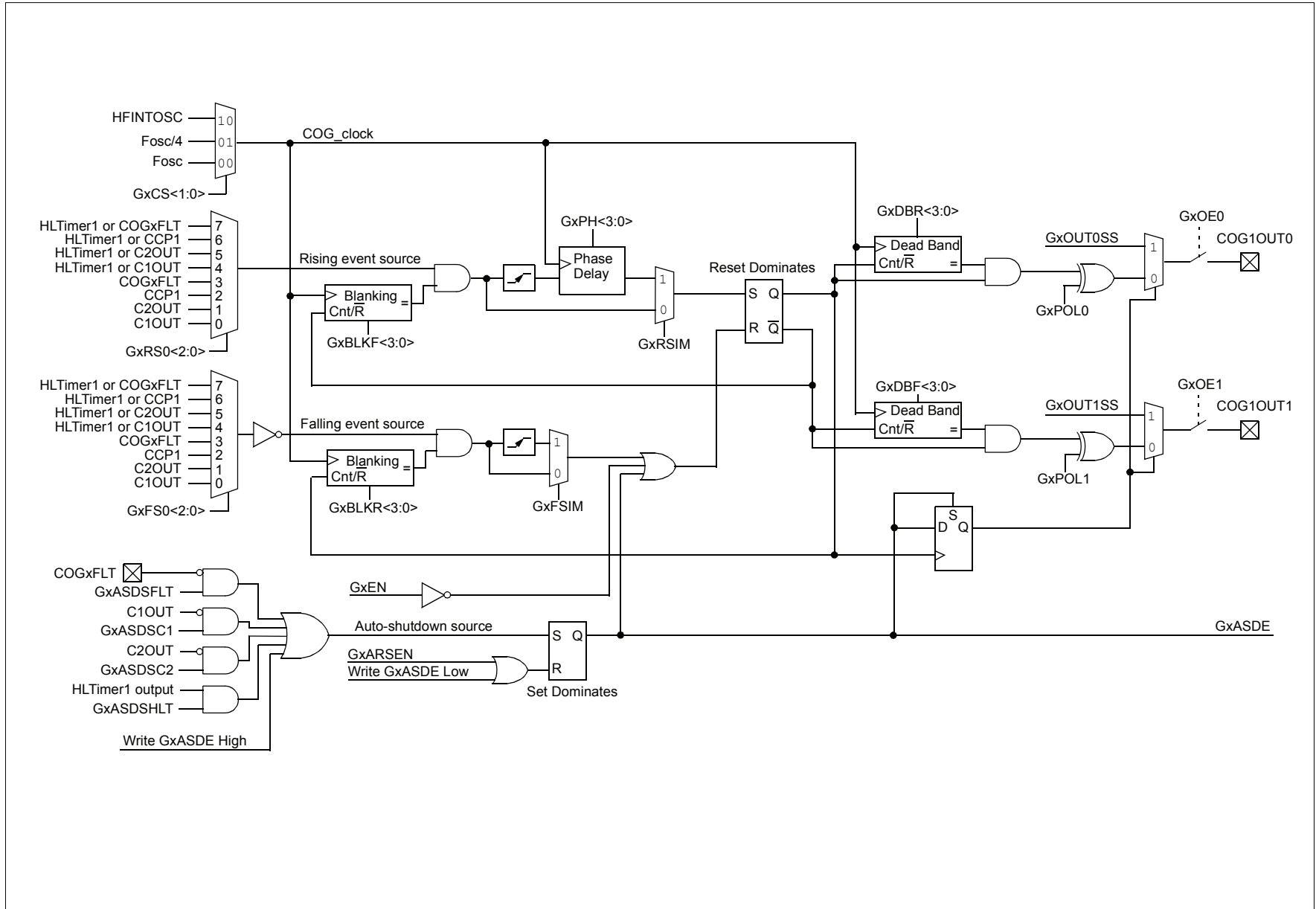


FIGURE 11-2: TYPICAL COG OPERATION WITH CCP1

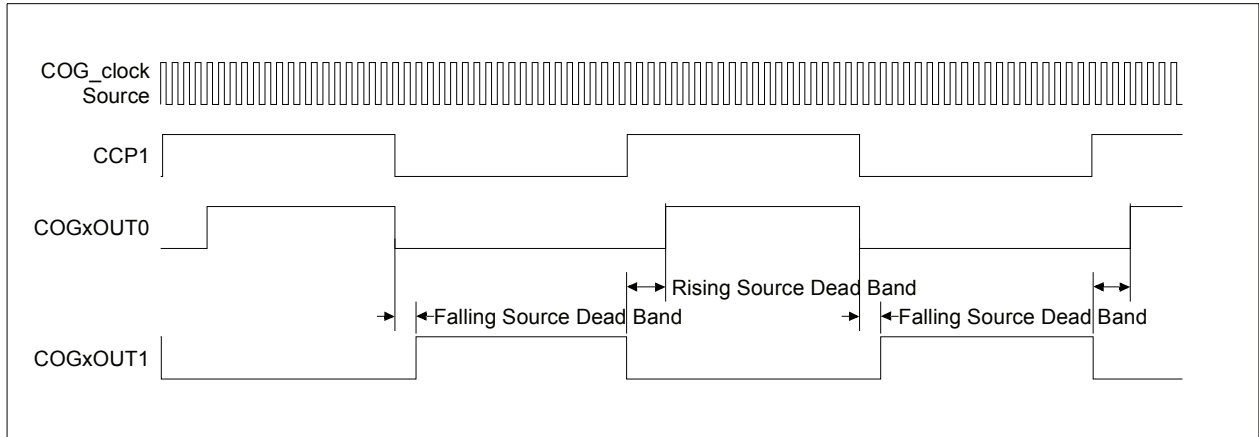
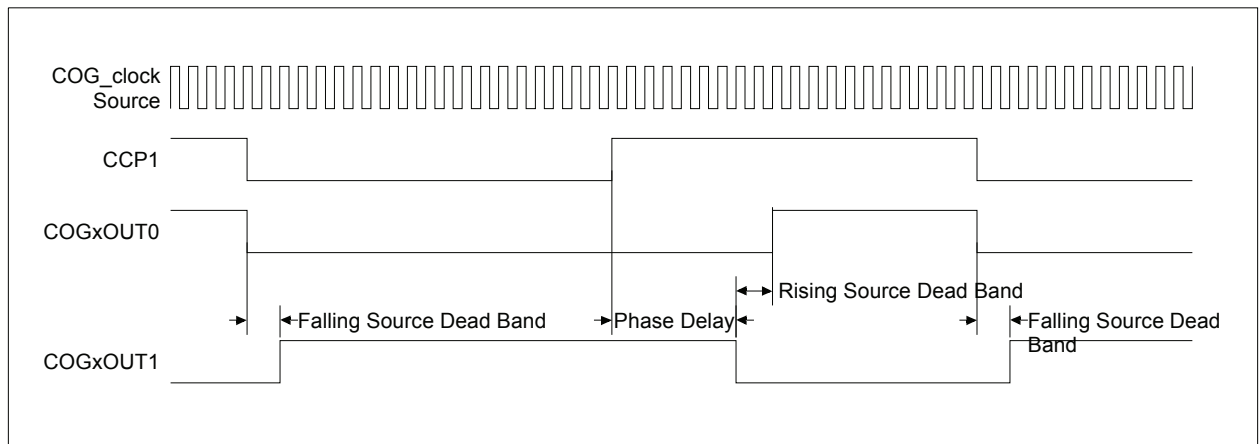


FIGURE 11-3: COG OPERATION WITH CCP1 AND PHASE DELAY



PIC12F752/HV752

11.2 Clock Sources

The COG_clock is used as the reference clock to the various timers in the peripheral. Timers that use the COG_clock include:

- Rising and falling dead-band time
- Rising and falling blanking time
- Rising event phase delay

Clock sources available for selection include:

- 8 MHz HFINTOSC
- Instruction clock (Fosc/4)
- System clock (Fosc)

The clock source is selected with the GxCS<1:0> bits of the COGxCON0 register ([Register 11-1](#)).

11.3 Selectable Event Sources

The COG uses two independently selectable event sources to generate the complementary waveform:

- Rising event source
- Falling event source

Level or edge sensitive modes are available for each event input.

The rising event source is selected with the GxRS<2:0> bits and the mode is controlled with the GxRSIM bit. The falling event source is selected with the GxFS<2:0> bits and the mode is controlled with the GxFSIM bit. Selection and mode control bits for both sources are located in the COGxCON1 register ([Register 11-2](#)).

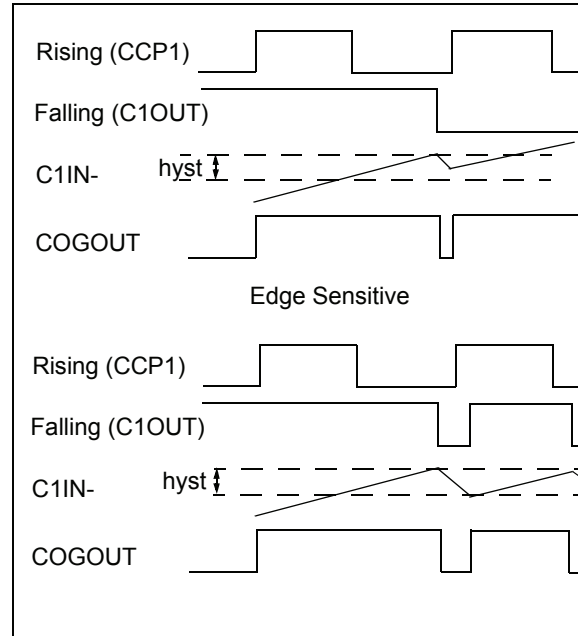
11.3.1 EDGE VS. LEVEL SENSING

Event input detection may be selected as level or edge sensitive. In general, events that are driven from a periodic source should be edge detected and events that are derived from voltage thresholds at the target circuit should be level sensitive. Consider the following two examples:

1. The first example is an application in which the period is determined by a 50% duty cycle clock and the COG output duty cycle is determined by a voltage level fed back through a comparator. If the clock input is level sensitive then duty cycles less than 50% will exhibit erratic operation.

2. The second example is similar to the first except that the duty cycle is close to 100%. The feedback comparator high-to-low transition trips the COG drive off but almost immediately the period source turns the drive back on. If the off cycle is short enough then the comparator input may not reach the low side of the hysteresis band precluding an output change. The comparator output stays low and without a high-to-low transition to trigger the edge sense then the drive of the COG output will be stuck in a constant drive-on condition. See [Figure 11-4](#).

FIGURE 11-4: EDGE VS LEVEL SENSE



11.3.2 RISING EVENT

The rising event starts the PWM output active duty cycle period. The rising event is the low-to-high transition of the selected rising event source. When the phase delay and rising event dead-band time values are zero, the COGxOUT0 output starts immediately. Otherwise, the COGxOUT0 output is delayed. The rising event causes all the following actions:

- Start rising event phase delay counter (if enabled).
- Clear COGxOUT1 after phase delay.
- Start falling event input blanking (if enabled).
- Start dead-band counter (if enabled).
- Set COGxOUT0 output after dead-band counter expires.

11.3.3 FALLING EVENT

The falling event terminates the PWM output active duty cycle period. The falling event is the high-to-low transition of the selected falling event source. When the falling event dead-band time value is zero, the COGxOUT1 output starts immediately. Otherwise, the COGxOUT1 output is delayed. The falling event causes all the following actions:

- Clear COGxOUT0.
- Start rising event input blanking (if enabled).
- Start falling event dead-band counter (if enabled).
- Set COGxOUT1 output after dead-band counter expires.

11.4 Output Control

Immediately after the COG module is enabled, the complementary drive is configured with COGxOUT0 drive cleared and COGxOUT1 drive active.

11.4.1 OUTPUT ENABLES

Each COG output pin has individual output enable controls. Output enables are selected with the GxOE0 and GxOE1 bits of the COGxCON0 register. When an output enable control is cleared, the module asserts no control over the pin. When an output enable is set, the override value or active PWM waveform is applied to the pin per the port priority selection.

The output pin enables are independent of the module enable bit, GxEN. When GxEN is cleared, the shutdown override levels are present on the COG output pins for which the output enables are active.

11.4.2 POLARITY CONTROL

The polarity of each COG output can be selected independently. When the output polarity bit is set, the corresponding output is active low. Clearing the output polarity bit configures the corresponding output as active high. However, polarity does not affect the override levels.

Output polarity is selected with the GxPOL0 and GxPOL1 bits of the COGxCON0 register.

11.5 Dead-Band Control

The dead-band control provides for non-overlapping PWM output signals to prevent shoot through current in the external power switches.

The COG contains two 4-bit dead-band counters. One dead-band counter is used for rising event dead-band control. The other is used for falling event dead-band control.

Dead band is timed by counting COG_clock periods from zero up to the value in the dead-band count register. Use [Equation 11-1](#) to calculate dead-band times.

11.5.1 RISING EVENT DEAD BAND

Rising event dead-band delays the turn-on of COGxOUT0 from when COGxOUT1 is turned off. The rising event dead-band time starts when the rising event output goes true.

The rising event output into the dead-band counter may be delayed by the phase delay. When the phase delay time is zero, the rising event output goes true coincident with the unblanked rising input event. When the phase delay time is not zero, the rising event output goes true at the completion of the phase delay time.

The rising event dead-band time is set by the value contained in the GxDBR<3:0> bits of the COGxDB register. When the value is zero, rising event dead band is disabled.

11.5.2 FALLING EVENT DEAD BAND

Falling event dead-band delays the turn-on of COGxOUT1 from when COGxOUT0 is turned off. The falling event dead-band time starts when the falling event output goes true. The falling event output goes true coincident with the unblanked falling input event.

The falling event dead-band time is set by the value contained in the GxDBF<3:0> bits of the COGxDB register. When the value is zero, falling event dead band is disabled.

11.5.3 DEAD-BAND TIME UNCERTAINTY

When the rising and falling events that trigger the dead-band counters come from asynchronous inputs, it creates uncertainty in the dead-band time. The maximum uncertainty is equal to one COG_clock period. Refer to [Equation 11-1](#) for more detail.

11.5.4 DEAD-BAND OVERLAP

There are two cases of dead-band overlap:

- Rising-to-falling
- Falling-to-rising

11.5.4.1 Rising-to-Falling Overlap

In this case, the falling event occurs while the rising event dead-band counter is still counting. When this happens, the COGxOUT0 drive is suppressed and the dead band extends by the falling event dead-band time. At the termination of the extended dead-band time, the COGxOUT1 drive goes true.

11.5.4.2 Falling-to-Rising Overlap

In this case, the rising event occurs while the falling event dead-band counter is still counting. When this happens, the COGxOUT1 drive is suppressed and the dead band extends by the rising event dead-band time. At the termination of the extended dead-band time, the COGxOUT0 drive goes true.

11.6 Blanking Control

Input blanking is a function whereby the event inputs can be masked or blanked for a short period of time. This is to prevent electrical transients caused by the turn-on/off of power components from generating a false input event.

The COG contains two 4-bit blanking counters. The counters are cross coupled with the events they are blanking. The falling event blanking counter is used to blank rising input events and the rising event blanking counter is used to blank falling input events. Once started, blanking extends for the time specified by the corresponding blanking counter.

Blanking is timed by counting COG_clock periods from zero up to the value in the blanking count register. Use Equation 11-1 to calculate blanking times.

11.6.1 RISING EVENT INPUT BLANKING

The falling event blanking counter inhibits the rising input from triggering a rising event. The falling event blanking time starts when the falling event output drive goes true.

The falling event blanking time is set by the value contained in the GxBLKF<3:0> bits of the COGxBLK register. Blanking times are calculated using the formula shown in Equation 11-1.

When the GxBLKF<3:0> value is zero, falling event blanking is disabled and the blanking counter output is true, thereby, allowing the event signal to pass straight through to the event trigger circuit.

11.6.2 FALLING EVENT INPUT BLANKING

The rising event blanking counter inhibits the falling input from triggering a falling event. The rising event blanking time starts when the rising event output drive goes true.

The rising event blanking time is set by the value contained in the GxBLKR<3:0> bits of the COGxBLK register.

When the GxBLKR<3:0> value is zero, rising event blanking is disabled and the blanking counter output is true, thereby, allowing the event signal to pass straight through to the event trigger circuit.

11.6.3 BLANKING TIME UNCERTAINTY

When the rising and falling events that trigger the blanking counters are asynchronous to the COG_clock, it creates uncertainty in the blanking time. The maximum uncertainty is equal to one COG_clock period. Refer to Equation 11-1 and Example 11-1 for more detail.

11.7 Phase Delay

It is possible to delay the assertion of the rising event. This is accomplished by placing a non-zero value in COGxPH register. Refer to Register 11-6 and Figure 11-3 for COG operation with CCP1 and phase delay. The delay from the input rising event signal switching to the actual assertion of the events is calculated the same as the dead-band and blanking delays. Please see Equation 11-1.

When the COGxPH value is zero, phase delay is disabled and the phase delay counter output is true, thereby, allowing the event signal to pass straight through to complementary output driver flop.

11.7.1 CUMULATIVE UNCERTAINTY

It is not possible to create more than one COG_clock of uncertainty by successive stages. Consider that the phase delay stage comes after the blanking stage, the dead-band stage comes after either the blanking or phase delay stages, and the blanking stage comes after the dead-band stage. When the preceding stage is enabled, the output of that stage is necessarily synchronous with the COG_clock, which removes any possibility of uncertainty in the succeeding stage.

EQUATION 11-1: PHASE, DEAD-BAND, AND BLANKING TIME CALCULATION

$$T_{\min} = \frac{\text{Count}}{F_{\text{COG_clock}}}$$

$$T_{\max} = \frac{\text{Count} + 1}{F_{\text{COG_clock}}}$$

$$T_{\text{uncertainty}} = T_{\max} - T_{\min}$$

Also:

$$T_{\text{uncertainty}} = \frac{1}{F_{\text{COG_clock}}}$$

Where:

| T | Count |
|------------------------|-------------|
| Phase Delay | GxPH<3:0> |
| Rising Dead Band | GxDBR<3:0> |
| Falling Dead Band | GxDBF<3:0> |
| Rising Event Blanking | GxBLKR<3:0> |
| Falling Event Blanking | GxBLKF<3:0> |

EXAMPLE 11-1: TIMER UNCERTAINTY

Given:

$$\begin{aligned}Count &= Ah = 10d \\F_{COG_Clock} &= 8\text{MHz}\end{aligned}$$

Therefore:

$$\begin{aligned}T_{\text{uncertainty}} &= \frac{1}{F_{COG_clock}} \\&= \frac{1}{8\text{MHz}} = 125\text{ns}\end{aligned}$$

Proof:

$$\begin{aligned}T_{\text{min}} &= \frac{Count}{F_{COG_clock}} \\&= 125\text{ns} \cdot 10d = 1.25\mu\text{s}\end{aligned}$$

$$\begin{aligned}T_{\text{max}} &= \frac{Count + 1}{F_{COG_clock}} \\&= 125\text{ns} \cdot (10d + 1) \\&= 1.375\mu\text{s}\end{aligned}$$

Therefore:

$$\begin{aligned}T_{\text{uncertainty}} &= T_{\text{max}} - T_{\text{min}} \\&= 1.375\mu\text{s} - 1.25\mu\text{s} \\&= 125\text{ns}\end{aligned}$$

PIC12F752/HV752

11.8 Auto-shutdown Control

Auto-shutdown is a method to immediately override the COG output levels with specific overrides that allow for safe shutdown of the circuit.

The shutdown state can be either cleared automatically or held until cleared by software.

11.8.1 SHUTDOWN

The shutdown state can be entered by either of the following two mechanisms:

- Software generated
- External Input

11.8.1.1 Software Generated Shutdown

Setting the GxASDE bit of the COGxASD register will force the COG into the shutdown state.

When auto-restart is disabled, the shutdown state will persist as long as the GxASDE bit is set.

When auto-restart is enabled, the GxASDE bit will clear automatically and resume operation on the next rising event. See [Figure 11-5](#).

11.8.1.2 External Shutdown Source

External shutdown inputs provide the fastest way to safely suspend COG operation in the event of a fault condition. When any of the selected shutdown inputs goes true, the output drive latches are reset and the COG outputs will immediately go to the selected override levels without software delay.

Any combination of four input sources can be selected to cause a shutdown condition. The four sources include:

- HLTimer1 output
- C2OUT (low true)
- C1OUT (low true)
- COG1FLT pin (low true)

Shutdown inputs are selected independently with bits <3:0> of the COGxASD register ([Register 11-3](#)).

Note: Shutdown inputs are level sensitive, not edge sensitive. The shutdown state cannot be cleared as long as the shutdown input level persists, except by disabling auto-shutdown,

11.8.2 PIN OVERRIDE LEVELS

The levels driven to the output pins, while the shutdown input is true, are controlled by the GxASDL0 and GxASDL1 bits of the COGxASD register ([Register 11-3](#)). GxASDL0 controls the GxOUT0 override level and GxASDL1 controls the GxOUT1 override level. The control bit logic level corresponds to the output logic drive level while in the shutdown state.

Note: The polarity control does not apply to the override level.

11.8.3 AUTO-SHUTDOWN RESTART

After an auto-shutdown event has occurred, there are two ways to have the module resume operation:

- Software controlled
- Auto-restart

The restart method is selected with the GxARSEN bit of the COGxASD register. Waveforms of a software controlled automatic restart are shown in [Figure 11-5](#).

11.8.3.1 Software Controlled Restart

When the GxARSEN bit of the COGxASD register is cleared, the COG must be restarted after an auto-shutdown event by software.

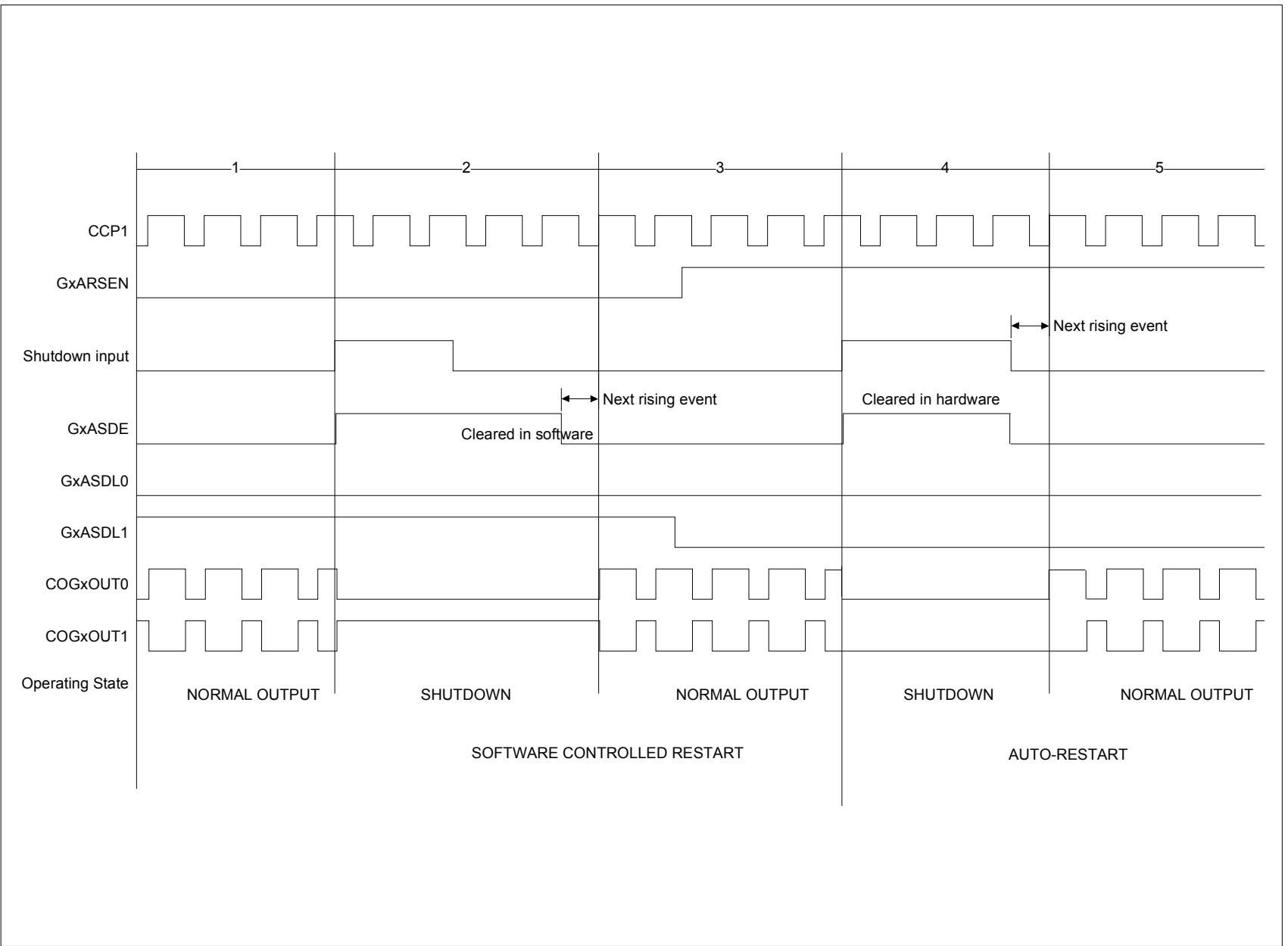
The COG will resume operation on the first rising event after the GxASDE bit is cleared. Clearing the shutdown state requires all selected shutdown inputs to be false, otherwise, the GxASDE bit will remain set.

11.8.3.2 Auto-Restart

When the GxARSEN bit of the COGxASD register is set then the COG will restart from the auto-shutdown state automatically.

The GxASDE bit will clear automatically and the COG will resume operation on the first rising event after all selected shutdown inputs go false.

FIGURE 11-5: AUTO-SHUTDOWN WAVEFORM – CCP1 AS RISING AND FALLING EVENT INPUT SOURCE



PIC12F752/HV752

11.9 Buffer updates

Changes to the phase, dead band, and blanking count registers need to occur simultaneously during COG operation to avoid unintended operation that may occur as a result of delays between each register write. This is accomplished with the GxLD bit of the COGxCON0 register and double buffering of the phase, blanking, and dead-band count registers.

Before the COG module is enabled, writing the count registers loads the count buffers without need of the GxLD bit. However, when the COG is enabled, the count buffers updates are suspended after writing the count registers until after the GxLD bit is set. When the GxLD bit is set, the phase, dead band, and blanking register values are transferred to the corresponding buffers synchronous with COG operation. The GxLD bit is cleared by hardware when the transfer is complete.

11.10 Alternate Pin Selection

The COGxOUT0, COGxOUT1 and COGxFLT functions can be directed to alternate pins with control bits of the APFCON register. Refer to [Register 5-1](#).

Note: The default COG outputs have high drive strength capability, whereas the alternate outputs do not.

11.11 Operation During Sleep

The COG continues to operate in Sleep provided that the COG_clock, rising event, and falling event sources remain active.

The HFINTSOC remains active during Sleep when the COG is enabled and the HFINTOSC is selected as the COG_clock source.

11.12 Configuring the COG

The following steps illustrate how to properly configure the COG to ensure a synchronous start with the rising event input:

1. Configure the desired COGxFLT input, COGxOUT0 and COGxOUT1 pins with the corresponding bits in the APFCON register.
2. Clear all ANSELA register bits associated with pins that are used for COG functions.
3. Ensure that the TRIS control bits corresponding to COGxOUT0 and COGxOUT1 are set so that both are configured as inputs. These will be set as outputs later.
4. Clear the GxEN bit, if not already cleared.
5. Set desired dead-band times with the COGxDB register.
6. Set desired blanking times with the COGxBLK register.
7. Set desired phase delay with the COGxPH register.
8. Setup the following controls in COGxASD auto-shutdown register:
 - Select desired shutdown sources.
 - Select both output overrides to the desired levels (this is necessary, even if not using auto-shutdown because start-up will be from a shutdown state).
 - Set the GxASDE bit and clear the GxARSEN bit.
9. Select the desired rising and falling event sources and input modes with the COGxCON1 register.
10. Configure the following controls in COGxCON0 register:
 - Select the desired clock source
 - Select the desired output polarities
 - Set the output enables of the outputs to be used.
11. Set the GxEN bit.
12. Clear TRIS control bits corresponding to COGxOUT0 and COGxOUT1 to be used, thereby configuring those pins as outputs.
13. If auto-restart is to be used, set the GxARSEN bit and the GxASDE will be cleared automatically. Otherwise, clear the GxASDE bit to start the COG.

11.13 COG Control Registers

REGISTER 11-1: COGxCON0: COG CONTROL REGISTER 0

| | | | | | | | |
|---------|---------|---------|---------|---------|---------|-----------|---------|
| R/W-0/0 | R/W-0/0 | R/W-0/0 | R/W-0/0 | R/W-0/0 | R/W-0/0 | R/W-0/0 | R/W-0/0 |
| GxEN | GxOE1 | GxOE0 | GxPOL1 | GxPOL0 | GxLD | GxCS<1:0> | |
| bit 7 | | | | | | | bit 0 |

Legend:

| | | |
|----------------------|----------------------|---|
| R = Readable bit | W = Writable bit | U = Unimplemented bit, read as '0' |
| u = Bit is unchanged | x = Bit is unknown | -n/n = Value at POR and BOR/Value at all other Resets |
| '1' = Bit is set | '0' = Bit is cleared | q = Value depends on condition |

- bit 7 **GxEN:** COGx Enable bit
 1 = Module is enabled
 0 = Module is disabled

- bit 6 **GxOE1:** COGxOUT1 Output Enable bit
 1 = COGxOUT1 is available on associated I/O pin
 0 = COGxOUT1 is not available on associated I/O pin

- bit 5 **GxOE0:** COGxOUT0 Output Enable bit
 1 = COGxOUT0 is available on associated I/O pin
 0 = COGxOUT0 is not available on associated I/O pin

- bit 4 **GxPOL1:** COGxOUT1 Output Polarity bit
 1 = Output is inverted polarity
 0 = Output is normal polarity

- bit 3 **GxPOL0:** COGxOUT0 Output Polarity bit
 1 = Output is inverted polarity
 0 = Output is normal polarity

- bit 2 **GxLD:** COGx Load Buffers bit
 1 = Phase, blanking, and dead-band buffers to be loaded with register values on next input events
 0 = Register to buffer transfer is complete

- bit 1-0 **GxCS<1:0>:** COGx Clock Source Select bits
 11 = Reserved
 10 = 8 MHz HFINTOSC clock
 01 = Instruction clock (Fosc/4)
 00 = System clock (Fosc)

PIC12F752/HV752

REGISTER 11-2: COG_xCON1: COG CONTROL REGISTER 1

| | | | | | | | |
|---------|---------|-----------|---------|---------|-----------|---------|---------|
| R/W-0/0 | R/W-0/0 | R/W-0/0 | R/W-0/0 | R/W-0/0 | R/W-0/0 | R/W-0/0 | R/W-0/0 |
| GxFSIM | GxRSIM | GxFS<2:0> | | | GxRS<2:0> | | |
| bit 7 | | | | | | | bit 0 |

Legend:

| | | |
|----------------------|----------------------|---|
| R = Readable bit | W = Writable bit | U = Unimplemented bit, read as '0' |
| u = Bit is unchanged | x = Bit is unknown | -n/n = Value at POR and BOR/Value at all other Resets |
| '1' = Bit is set | '0' = Bit is cleared | q = Value depends on condition |

bit 7 **GxFSIM:** COG_x Falling Source Input mode bit

1 = Input is edge sensitive

0 = Input is level sensitive

bit 6 **GxRSIM:** COG_x Rising Source Input mode bit

1 = Input is edge sensitive

0 = Input is level sensitive

bit 5-3 **GxFS<2:0>:** COG_x Falling Source Select bits

111 = COG_xFLT or HLTimer1

110 = CCP1 or HLTimer1

101 = C2OUT or HLTimer1

100 = C1OUT or HLTimer1

011 = COG_xFLT

010 = CCP1

001 = C2OUT

000 = C1OUT

bit 2-0 **GxRS<2:0>:** COG_x Rising Source Select bits

111 = COG_xFLT or HLTimer1

110 = CCP1 or HLTimer1

101 = C2OUT or HLTimer1

100 = C1OUT or HLTimer1

011 = COG_xFLT

010 = CCP1

001 = C2OUT

000 = C1OUT

REGISTER 11-3: COGxASD: COG AUTO-SHUTDOWN CONTROL REGISTER

| | | | | | | | |
|---------|---------|---------|---------|-----------|----------|----------|-----------|
| R/W-0/0 | R/W-0/0 | R/W-0/0 | R/W-0/0 | R/W-0/0 | R/W-0/0 | R/W-0/0 | R/W-0/0 |
| GxASDE | GxARSEN | GxASDL1 | GxASDL0 | GxASDSHLT | GxASDSC2 | GxASDSC1 | GxASDSFLT |
| bit 7 | | | | | | | bit 0 |

Legend:

| | | |
|----------------------|----------------------|---|
| R = Readable bit | W = Writable bit | U = Unimplemented bit, read as '0' |
| u = Bit is unchanged | x = Bit is unknown | -n/n = Value at POR and BOR/Value at all other Resets |
| '1' = Bit is set | '0' = Bit is cleared | q = Value depends on condition |

- bit 7 **GxASDE:** Auto-Shutdown Event Status bit
1 = COG is in the shutdown state
0 = COG is not in the shutdown state
- bit 6 **GxARSEN:** Auto-Restart Enable bit
1 = Auto-restart is enabled
0 = Auto-restart is disabled
- bit 5 **GxASDL1:** COGxOUT1 Auto-shutdown Override Level bit
1 = A logic '1' is placed on COGxOUT1 when a shutdown input is true
0 = A logic '0' is placed on COGxOUT1 when a shutdown input is true
- bit 4 **GxASDL0:** COGxOUT0 Auto-shutdown Override Level bit
1 = A logic '1' is placed on COGxOUT0 when a shutdown input is true
0 = A logic '0' is placed on COGxOUT0 when a shutdown input is true
- bit 3 **GxASDSHLT:** COG Auto-shutdown Source Enable bit 3
1 = COG is shutdown when HLTMR equals HLTMR is low
0 = HLTMR1 pin has no effect on shutdown
- bit 2 **GxASDSC2:** COG Auto-shutdown Source Enable bit 2
1 = COG is shutdown when C2OUT is low
0 = C2OUT pin has no effect on shutdown
- bit 1 **GxASDSC1:** COG Auto-shutdown Source Enable bit 1
1 = COG is shutdown when C1OUT is low
0 = C1OUT pin has no effect on shutdown
- bit 0 **GxASDSFLT:** COG Auto-shutdown Source Enable bit 0
1 = COG is shutdown when COGxFLT pin is low
0 = COGxFLT pin has no effect on shutdown

PIC12F752/HV752

REGISTER 11-4: COGxDB: COG DEAD-BAND COUNT REGISTER

| | | | | | | | |
|------------|---------|---------|---------|------------|---------|---------|---------|
| R/W-x/u | R/W-x/u | R/W-x/u | R/W-x/u | R/W-x/u | R/W-x/u | R/W-x/u | R/W-x/u |
| GxDBR<3:0> | | | | GxDBF<3:0> | | | |
| bit 7 | | | | bit 0 | | | |

Legend:

R = Readable bit W = Writable bit U = Unimplemented bit, read as '0'
u = Bit is unchanged x = Bit is unknown -n/n = Value at POR and BOR/Value at all other Resets
'1' = Bit is set '0' = Bit is cleared q = Value depends on condition

bit 7-4 **GxDBR<3:0>**: Rising Event Dead-band Count Value bits
= Number of COG clock periods to delay primary output after rising event input
bit 3-0 **GxDBF<3:0>**: Falling Event Dead-band Count Value bits
= Number of COG clock periods to delay complementary output after falling event input

REGISTER 11-5: COGxBLK: COG BLANKING COUNT REGISTER

| | | | | | | | |
|-------------|---------|---------|---------|-------------|---------|---------|---------|
| R/W-x/u | R/W-x/u | R/W-x/u | R/W-x/u | R/W-x/u | R/W-x/u | R/W-x/u | R/W-x/u |
| GxBLKR<3:0> | | | | GxBLKF<3:0> | | | |
| bit 7 | | | | bit 0 | | | |

Legend:

R = Readable bit W = Writable bit U = Unimplemented bit, read as '0'
u = Bit is unchanged x = Bit is unknown -n/n = Value at POR and BOR/Value at all other Resets
'1' = Bit is set '0' = Bit is cleared q = Value depends on condition

bit 7-4 **GxBLKR<3:0>**: Rising Event Blanking Count Value bits
= Number of COGx clock periods to inhibit falling event input
bit 3-0 **GxBLKF<3:0>**: Falling Event Blanking Count Value bits
= Number of COGx clock periods to inhibit rising event input

REGISTER 11-6: COGxPH: COG PHASE COUNT REGISTER

| | | | | | | | |
|-------|-----|-----|-----|-----------|---------|---------|---------|
| U-0 | U-0 | U-0 | U-0 | R/W-x/u | R/W-x/u | R/W-x/u | R/W-x/u |
| — | — | — | — | GxPH<3:0> | | | |
| bit 7 | | | | bit 0 | | | |

Legend:

R = Readable bit W = Writable bit U = Unimplemented bit, read as '0'
u = Bit is unchanged x = Bit is unknown -n/n = Value at POR and BOR/Value at all other Resets
'1' = Bit is set '0' = Bit is cleared q = Value depends on condition

bit 7-4 **Unimplemented**: Read as '0'
bit 3-0 **GxPH<3:0>**: Rising Event Phase Delay Count Value bits
= Number of COG clock periods to delay rising edge event

TABLE 11-1: SUMMARY OF REGISTERS ASSOCIATED WITH COG

| Name | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | Register on Page |
|----------|-------------|---------|-----------|---------|-------------|-----------|-----------|-----------|------------------|
| ANSELA | — | — | ANSA5 | ANSA4 | — | ANSA2 | ANSA1 | ANSA0 | 50 |
| APFCON | — | — | — | T1GSEL | — | COG1FSEL | COG1O1SEL | COG1O0SEL | 46 |
| COG1PH | — | — | — | — | G1PH<3:0> | | | | 98 |
| COG1BLK | G1BLKR<3:0> | | | | G1BLKF<3:0> | | | | 98 |
| COG1DB | G1DBR<3:0> | | | | G1DBF<3:0> | | | | 98 |
| COG1CON0 | G1EN | G1OE1 | G1OE0 | G1POL1 | G1POL0 | G1LD | G1CS1 | G1CS0 | 95 |
| COG1CON1 | G1FSIM | G1RSIM | G1FS<2:0> | | | G1RS<2:0> | | | 96 |
| COG1ASD | G1ASDE | G1ARSEN | G1ASDL1 | G1ASDL0 | G1ASDShLT | G1ASDSC2 | G1ASDSC1 | G1ASDSFLT | 97 |
| INTCON | GIE | PEIE | T0IE | INTE | IOCIE | T0IF | INTF | IOCIF | 20 |
| LATA | — | — | LATA5 | LATA4 | — | LATA2 | LATA1 | LATA0 | 48 |
| PIE2 | — | — | C2IE | C1IE | — | COG1IE | — | CCP1IE | 22 |
| PIR2 | — | — | C2IF | C1IF | — | COG1IF | — | CCP1IF | 24 |
| TRISA | — | — | TRISA5 | TRISA4 | TRISA3 | TRISA2 | TRISA1 | TRISA0 | 48 |

Legend: x = unknown, u = unchanged, - = unimplemented locations read as '0'. Shaded cells are not used by COG.

- Note 1:** Other (non Power-up) Resets include MCLR Reset and Watchdog Timer Reset during normal operation.
Note 2: See Configuration Word register ([Register 17-1](#)) for operation of all register bits.

PIC12F752/HV752

NOTES:

12.0 ANALOG-TO-DIGITAL CONVERTER (ADC) MODULE

The Analog-to-Digital Converter (ADC) allows conversion of an analog input signal to a 10-bit binary representation of that signal. This device uses analog inputs, which are multiplexed into a single sample and hold circuit. The output of the sample and hold is connected to the input of the converter. The converter generates a 10-bit binary result via successive approximation and stores the conversion result into the ADC result registers (ADRESL and ADRESH).

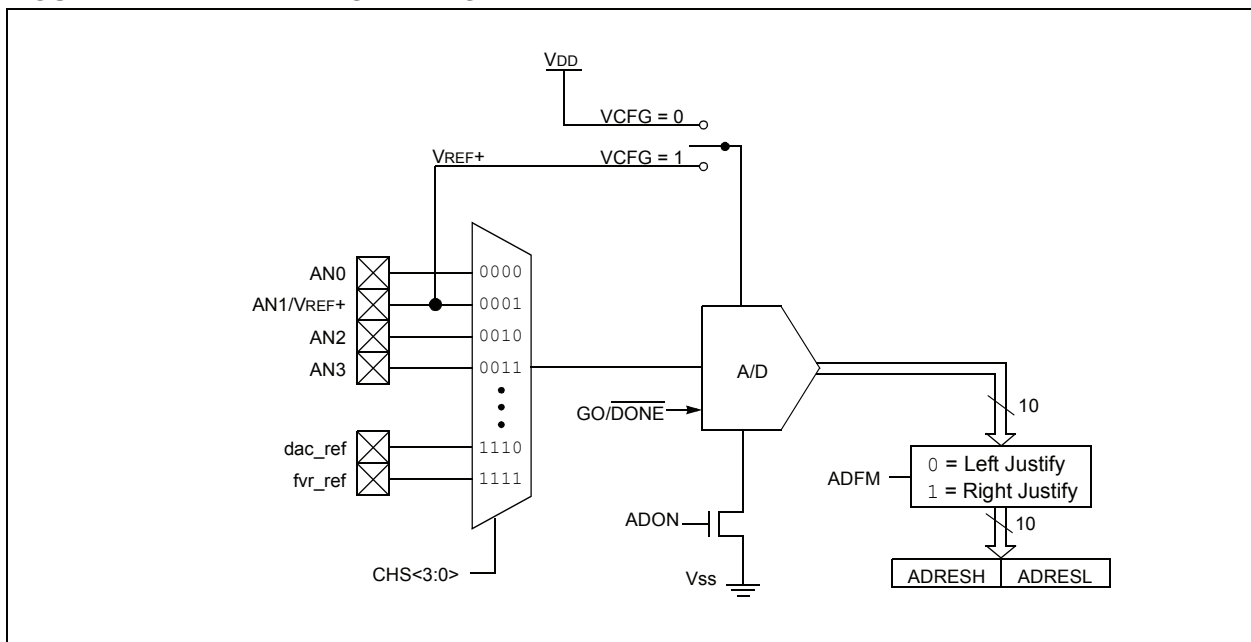
The ADC voltage reference is software selectable to either VDD or a voltage applied to the external reference pins.

The ADC can generate an interrupt upon completion of a conversion. This interrupt can be used to wake-up the device from Sleep.

Figure 12-1 shows the block diagram of the ADC.

Note: The ADRESL and ADRESH registers are read-only.

FIGURE 12-1: ADC BLOCK DIAGRAM



PIC12F752/HV752

12.1 ADC Configuration

When configuring and using the ADC the following functions must be considered:

- Port configuration
- Channel selection
- ADC voltage reference selection
- ADC conversion clock source
- Interrupt control
- Results formatting

12.1.1 PORT CONFIGURATION

The ADC can be used to convert both analog and digital signals. When converting analog signals, the I/O pin should be configured for analog by setting the associated TRIS and ANSEL bits. See the corresponding port section for more information.

Note: Analog voltages on any pin that is defined as a digital input may cause the input buffer to conduct excess current.

12.1.2 CHANNEL SELECTION

The CHS bits of the ADCON0 register determine which channel is connected to the sample and hold circuit.

When changing channels, a delay is required before starting the next conversion. Refer to [Section 12.2 “ADC Operation”](#) for more information.

12.1.3 ADC VOLTAGE REFERENCE

The VCFG bit of the ADCON0 register provides control of the positive voltage reference. The positive voltage reference can be either VDD or an external voltage source. The negative voltage reference is always connected to the ground reference.

12.1.4 CONVERSION CLOCK

The source of the conversion clock is software selectable via the ADCS bits of the ADCON1 register. There are seven possible clock options:

- FOSC/2
- FOSC/4
- FOSC/8
- FOSC/16
- FOSC/32
- FOSC/64
- FRC (dedicated internal oscillator)

The time to complete one bit conversion is defined as TAD. One full 10-bit conversion requires 11 TAD periods as shown in [Figure 12-2](#).

For correct conversion, the appropriate TAD specification must be met. See A/D conversion requirements in [Section 20.0 “Electrical Specifications”](#) for more information. [Table 12-1](#) gives examples of appropriate ADC clock selections.

Note: Unless using the FRC, any changes in the system clock frequency will change the ADC clock frequency, which may adversely affect the ADC result.

TABLE 12-1: ADC CLOCK PERIOD (TAD) VS. DEVICE OPERATING FREQUENCIES (VDD ≥ 3.0V)

| ADC Clock Period (TAD) | | Device Frequency (Fosc) | | | |
|------------------------|-----------|-------------------------|-------------------------|-------------------------|-------------------------|
| ADC Clock Source | ADCS<2:0> | 20 MHz | 8 MHz | 4 MHz | 1 MHz |
| Fosc/2 | 000 | 100 ns ⁽²⁾ | 250 ns ⁽²⁾ | 500 ns ⁽²⁾ | 2.0 μs |
| Fosc/4 | 100 | 200 ns ⁽²⁾ | 500 ns ⁽²⁾ | 1.0 μs ⁽²⁾ | 4.0 μs |
| Fosc/8 | 001 | 400 ns ⁽²⁾ | 1.0 μs ⁽²⁾ | 2.0 μs | 8.0 μs ⁽³⁾ |
| Fosc/16 | 101 | 800 ns ⁽²⁾ | 2.0 μs | 4.0 μs | 16.0 μs ⁽³⁾ |
| Fosc/32 | 010 | 1.6 μs | 4.0 μs | 8.0 μs ⁽³⁾ | 32.0 μs ⁽³⁾ |
| Fosc/64 | 110 | 3.2 μs | 8.0 μs ⁽³⁾ | 16.0 μs ⁽³⁾ | 64.0 μs ⁽³⁾ |
| FRC | x11 | 2-6 μs ^(1,4) | 2-6 μs ^(1,4) | 2-6 μs ^(1,4) | 2-6 μs ^(1,4) |

Legend: Shaded cells are outside of recommended range.

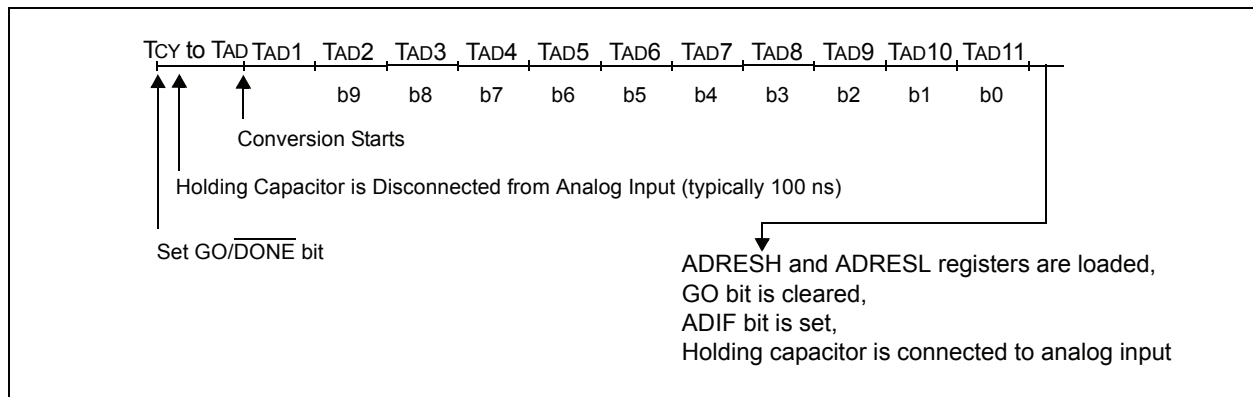
Note 1: The FRC source has a typical TAD time of 4 μs for VDD > 3.0V.

2: These values violate the minimum required TAD time.

3: For faster conversion times, the selection of another clock source is recommended.

4: When the device frequency is greater than 1 MHz, the FRC clock source is only recommended if the conversion will be performed during Sleep.

FIGURE 12-2: ANALOG-TO-DIGITAL CONVERSION TAD CYCLES



12.1.5 INTERRUPTS

The ADC module allows for the ability to generate an interrupt upon completion of an Analog-to-Digital conversion. The ADC interrupt flag is the ADIF bit in the PIR1 register. The ADC interrupt enable is the ADIE bit in the PIE1 register. The ADIF bit must be cleared in software.

Note: The ADIF bit is set at the completion of every conversion, regardless of whether or not the ADC interrupt is enabled.

This interrupt can be generated while the device is operating or while in Sleep. If the device is in Sleep, the interrupt will wake-up the device. Upon waking from Sleep, the next instruction following the SLEEP instruction is always executed. If the user is attempting to wake-up from Sleep and resume in-line code execution, the global interrupt must be disabled. If the global interrupt is enabled, execution will switch to the Interrupt Service Routine.

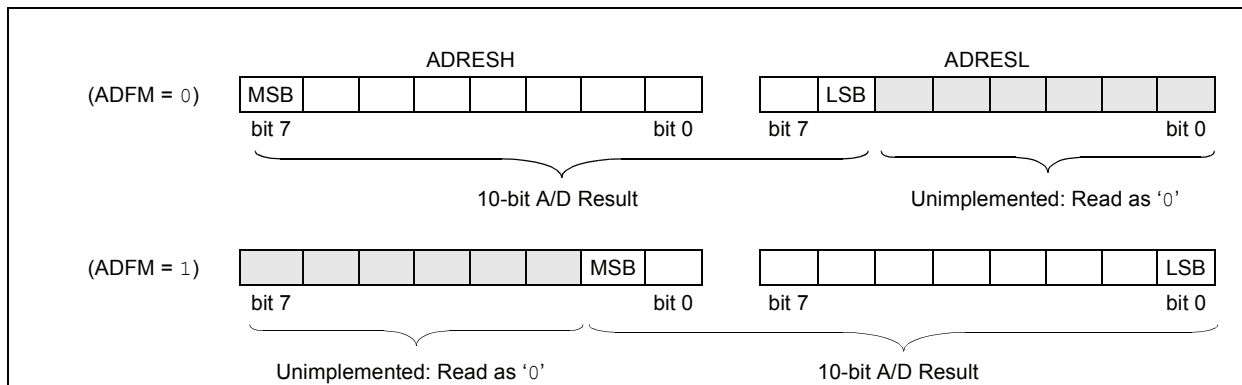
PIC12F752/HV752

12.1.6 RESULT FORMATTING

The 10-bit A/D conversion result can be supplied in two formats, left justified or right justified. The ADFM bit of the ADCON0 register controls the output format.

Figure 12-4 shows the two output formats.

FIGURE 12-3: 10-BIT A/D CONVERSION RESULT FORMAT



12.2 ADC Operation

12.2.1 STARTING A CONVERSION

To enable the ADC module, the ADON bit of the ADCON0 register must be set to a '1'. Setting the GO/DONE bit of the ADCON0 register to a '1' will start the Analog-to-Digital conversion.

Note: The GO/DONE bit should not be set in the same instruction that turns on the ADC. Refer to [Section 12.2.6 "A/D Conversion Procedure"](#).

12.2.2 COMPLETION OF A CONVERSION

When the conversion is complete, the ADC module will:

- Clear the GO/DONE bit
- Set the ADIF flag bit
- Update the ADRESH:ADRESL registers with new conversion result

12.2.3 TERMINATING A CONVERSION

If a conversion must be terminated before completion, the GO/DONE bit can be cleared in software. The ADRESH:ADRESL registers will not be updated with the partially complete Analog-to-Digital conversion sample. Instead, the ADRESH:ADRESL register pair will retain the value of the previous conversion. Additionally, a 2 TAD delay is required before another acquisition can be initiated. Following this delay, an input acquisition is automatically started on the selected channel.

Note: A device Reset forces all registers to their Reset state. Thus, the ADC module is turned off and any pending conversion is terminated.

12.2.4 ADC OPERATION DURING SLEEP

The ADC module can operate during Sleep. This requires the ADC clock source to be set to the FRC option. When the FRC clock source is selected, the ADC waits one additional instruction before starting the conversion. This allows the SLEEP instruction to be executed, which can reduce system noise during the conversion. If the ADC interrupt is enabled, the device will wake-up from Sleep when the conversion completes. If the ADC interrupt is disabled, the ADC module is turned off after the conversion completes, although the ADON bit remains set.

When the ADC clock source is something other than FRC, a SLEEP instruction causes the present conversion to be aborted and the ADC module is turned off, although the ADON bit remains set.

12.2.5 SPECIAL EVENT TRIGGER

The CCP Special Event Trigger allows periodic ADC measurements without software intervention. When this trigger occurs, the GO/DONE bit is set by hardware and the Timer1 counter resets to zero.

Using the Special Event Trigger does not assure proper ADC timing. It is the user's responsibility to ensure that the ADC timing requirements are met.

See [Section 10.0 "Capture/Compare/PWM Modules"](#) for more information.

12.2.6 A/D CONVERSION PROCEDURE

This is an example procedure for using the ADC to perform an Analog-to-Digital conversion:

1. Configure Port:
 - Disable pin output driver (See TRIS register)
 - Configure pin as analog
2. Configure the ADC module:
 - Select ADC conversion clock
 - Configure voltage reference
 - Select ADC input channel
 - Select result format
 - Turn on ADC module
3. Configure ADC interrupt (optional):
 - Clear ADC interrupt flag
 - Enable ADC interrupt
 - Enable peripheral interrupt
 - Enable global interrupt⁽¹⁾
4. Wait the required acquisition time⁽²⁾.
5. Start conversion by setting the GO/DONE bit.
6. Wait for ADC conversion to complete by one of the following:
 - Polling the GO/DONE bit
 - Waiting for the ADC interrupt (interrupts enabled)
7. Read ADC Result
8. Clear the ADC interrupt flag (required if interrupt is enabled).

EXAMPLE 12-1: A/D CONVERSION

```

;This code block configures the ADC
;for polling, Vdd reference, Frc clock
;and RA0 input.
;
;Conversion start & polling for completion
;are included.
;
BANKSEL TRISA      ;
BSF      TRISA,0   ;Set RA0 to input
BANKSEL ADCON1    ;
MOVLW   B'01110000' ;ADC Frc clock,
IORWF   ADCON1    ; and RA0 as analog
BANKSEL ADCON0    ;
MOVLW   B'10000001' ;Right justify,
MOVWF   ADCON0    ;Vdd Vref, AN0, On
CALL    SampleTime ;Acquisiton delay
BSF     ADCON0,GO  ;Start conversion
TEST AGAIN
BTFSC   ADCON0,GO  ;Is conversion done?
GOTO    TEST AGAIN ;No, test again
BANKSEL ADRESH    ;
MOVF    ADRESH,W  ;Read upper 2 bits
MOVWF   RESULTHI  ;Store in GPR space
BANKSEL ADRESL    ;
MOVF    ADRESL,W  ;Read lower 8 bits
MOVWF   RESULTLO  ;Store in GPR space
    
```

Note 1: The global interrupt can be disabled if the user is attempting to wake-up from Sleep and resume in-line code execution.

2: See [Section 12.4 “A/D Acquisition Requirements”](#).

PIC12F752/HV752

12.3 ADC Control Registers

REGISTER 12-1: ADCON0: A/D CONTROL REGISTER 0

| | | | | | | | |
|-------|-------|----------|-------|-------|-------|---------|-------|
| R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 |
| ADFM | VCFG | CHS<3:0> | | | | GO/DONE | ADON |
| bit 7 | | | | | | | bit 0 |

Legend:

R = Readable bit

W = Writable bit

U = Unimplemented bit, read as '0'

-n = Value at POR

'1' = Bit is set

'0' = Bit is cleared

x = Bit is unknown

bit 7 **ADFM:** A/D Conversion Result Format Select bit

1 = Right justified

0 = Left justified

bit 6 **VCFG:** Voltage Reference bit

1 = VREF pin

0 = VDD

bit 5-2 **CHS<3:0>:** Analog Channel Select bits

0000 = Channel 00 (AN0)

0001 = Channel 01 (AN1)

0010 = Channel 02 (AN2)

0011 = Channel 03 (AN3)

0100 = Reserved. Do not use.

•

•

•

1101 = Reserved. Do not use.

1110 = Digital-to-Analog Converter (DAC output)

1111 = Fixed Voltage Reference (FVR)

bit 1 **GO/DONE:** A/D Conversion Status bit

1 = A/D conversion cycle in progress. Setting this bit starts an A/D conversion cycle.

This bit is automatically cleared by hardware when the A/D conversion has completed.

0 = A/D conversion completed/not in progress

bit 0 **ADON:** ADC Enable bit

1 = ADC is enabled

0 = ADC is disabled and consumes no operating current

REGISTER 12-2: ADCON1: A/D CONTROL REGISTER 1

| | | | | | | | |
|-------|-----------|---------|---------|-----|-----|-----|-------|
| U-0 | R/W-0/0 | R/W-0/0 | R/W-0/0 | U-0 | U-0 | U-0 | U-0 |
| — | ADCS<2:0> | | | — | — | — | — |
| bit 7 | | | | | | | bit 0 |

Legend:

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

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

bit 6-4 **ADCS<2:0>:** A/D Conversion Clock Select bits

000 = Fosc/2

001 = Fosc/8

010 = Fosc/32

011 = FRC (clock supplied from an internal oscillator with a divisor of 16)

100 = Fosc/4

101 = Fosc/16

110 = Fosc/64

bit 3-0 **Unimplemented:** Read as '0'

PIC12F752/HV752

REGISTER 12-3: ADRESH: ADC RESULT REGISTER HIGH (ADRESH) ADFM = 0 (READ-ONLY)

| | | | | | | | |
|------------|-----|-----|-----|-----|-----|-----|-------|
| R-x | R-x | R-x | R-x | R-x | R-x | R-x | R-x |
| ADRES<9:2> | | | | | | | |
| bit 7 | | | | | | | bit 0 |

Legend:

R = Readable bit W = Writable bit U = Unimplemented bit, read as '0'
 -n = Value at POR '1' = Bit is set '0' = Bit is cleared x = Bit is unknown

bit 7-0 **ADRES<9:2>**: ADC Result Register bits
 Upper 8 bits of 10-bit conversion result

REGISTER 12-4: ADRESL: ADC RESULT REGISTER LOW (ADRESL) ADFM = 0 (READ-ONLY)

| | | | | | | | |
|------------|-----|-----|-----|-----|-----|-----|-------|
| R-x | R-x | U-0 | U-0 | U-0 | U-0 | U-0 | U-0 |
| ADRES<1:0> | | — | — | — | — | — | — |
| bit 7 | | | | | | | bit 0 |

Legend:

R = Readable bit W = Writable bit U = Unimplemented bit, read as '0'
 -n = Value at POR '1' = Bit is set '0' = Bit is cleared x = Bit is unknown

bit 7-6 **ADRES<1:0>**: ADC Result Register bits
 Lower 2 bits of 10-bit conversion result

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

REGISTER 12-5: ADRESH: ADC RESULT REGISTER HIGH (ADRESH) ADFM = 1 (READ-ONLY)

| | | | | | | | |
|-------|-----|-----|-----|-----|-----|------------|-----|
| U-0 | U-0 | U-0 | U-0 | U-0 | U-0 | R-x | R-x |
| — | — | — | — | — | — | ADRES<9:8> | |
| bit 7 | | | | | | bit 0 | |

Legend:

R = Readable bit W = Writable bit U = Unimplemented bit, read as '0'
 -n = Value at POR '1' = Bit is set '0' = Bit is cleared x = Bit is unknown

bit 7-2 **Unimplemented**: Read as '0'

bit 1-0 **ADRES<9:8>**: ADC Result Register bits
 Upper 2 bits of 10-bit conversion result

REGISTER 12-6: ADRESL: ADC RESULT REGISTER LOW (ADRESL) ADFM = 1 (READ-ONLY)

| | | | | | | | |
|------------|-----|-----|-----|-----|-----|-----|-------|
| R-x | R-x | R-x | R-x | R-x | R-x | R-x | R-x |
| ADRES<7:0> | | | | | | | |
| bit 7 | | | | | | | bit 0 |

Legend:

R = Readable bit W = Writable bit U = Unimplemented bit, read as '0'
 -n = Value at POR '1' = Bit is set '0' = Bit is cleared x = Bit is unknown

bit 7-0 **ADRES<7:0>**: ADC Result Register bits
 Lower 8 bits of 10-bit conversion result

12.4 A/D Acquisition Requirements

For the ADC to meet its specified accuracy, the charge holding capacitor (CHOLD) must be allowed to fully charge to the input channel voltage level. The Analog Input model is shown in Figure 12-4. The source impedance (RS) and the internal sampling switch (RSS) impedance directly affect the time required to charge the capacitor CHOLD. The sampling switch (RSS) impedance varies over the device voltage (VDD), see Figure 12-4. **The maximum recommended impedance for analog sources is 10 kΩ.** As the source impedance is decreased, the acquisition time may be decreased. After the analog input channel is selected (or changed),

an A/D acquisition must be done before the conversion can be started. To calculate the minimum acquisition time, Equation 12-1 may be used. This equation assumes that 1/2 LSB error is used (1024 steps for the ADC). The 1/2 LSB error is the maximum error allowed for the ADC to meet its specified resolution.

EQUATION 12-1: ACQUISITION TIME EXAMPLE

Assumptions: Temperature = 50°C and external impedance of 10kΩ 5.0V VDD

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

The value for TC can be approximated with the following equations:

$$V_{APPLIED} \left(1 - \frac{1}{2047} \right) = V_{CHOLD} \quad ;[1] \text{ } V_{CHOLD} \text{ charged to within } 1/2 \text{ lsb}$$

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

$$V_{APPLIED} \left(1 - e^{-\frac{T_C}{RC}} \right) = V_{APPLIED} \left(1 - \frac{1}{2047} \right) \quad ;\text{combining [1] and [2]}$$

Solving for TC:

$$\begin{aligned} T_C &= -CHOLD(RIC + RSS + RS) \ln(1/2047) \\ &= -10pF(1k\Omega + 7k\Omega + 10k\Omega) \ln(0.0004885) \\ &= 1.37\mu s \end{aligned}$$

Therefore:

$$\begin{aligned} T_{ACQ} &= 2\mu s + 1.37\mu s + [(50^\circ C - 25^\circ C)(0.05\mu s/^\circ C)] \\ &= 4.67\mu s \end{aligned}$$

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

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

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

PIC12F752/HV752

FIGURE 12-4: ANALOG INPUT MODEL

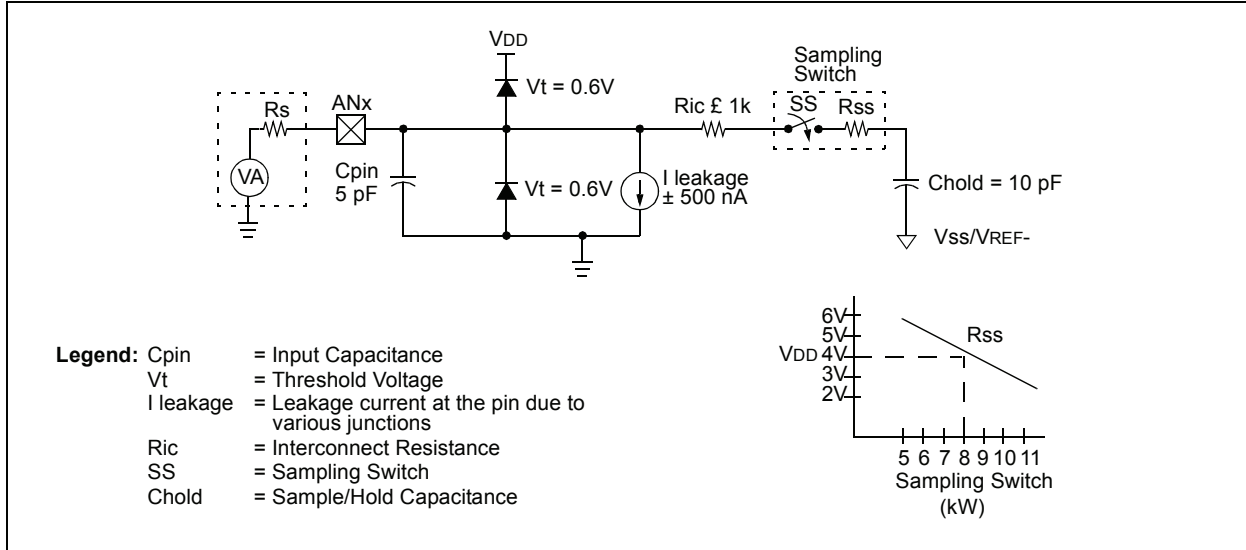


FIGURE 12-5: ADC TRANSFER FUNCTION

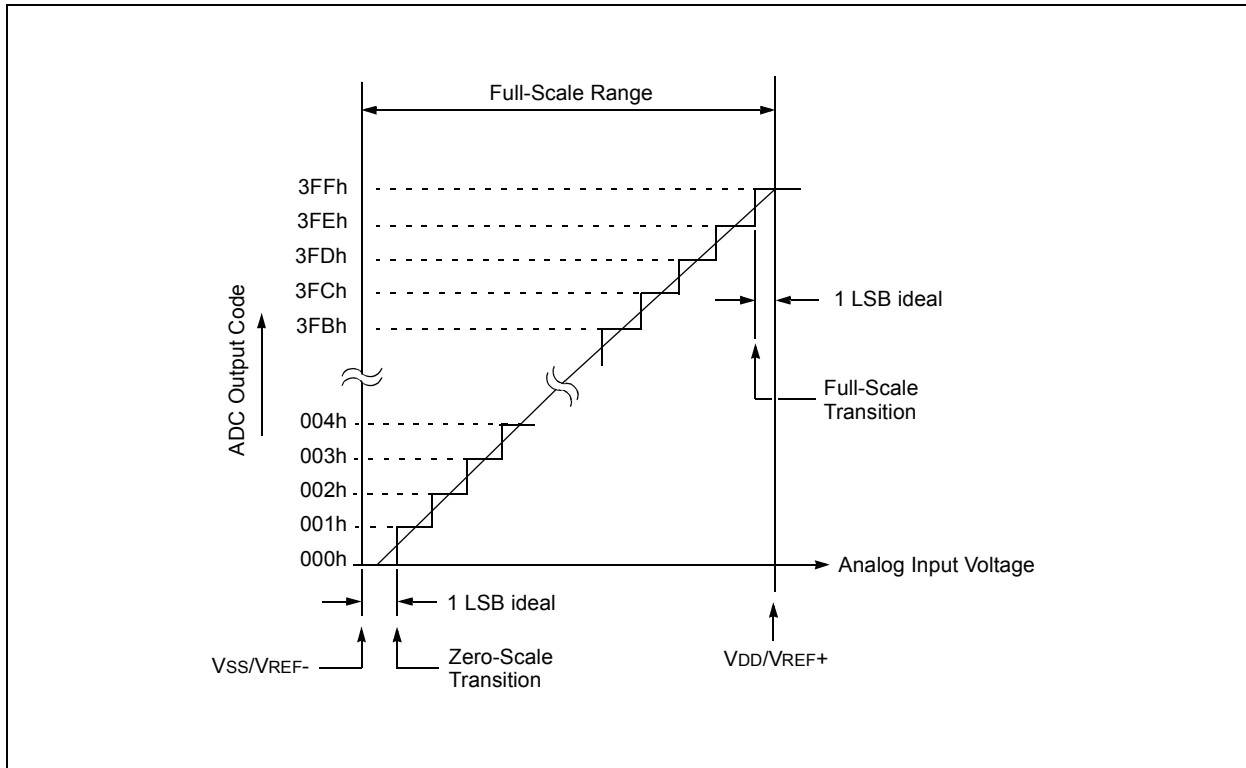


TABLE 12-2: SUMMARY OF ASSOCIATED ADC REGISTERS

| Name | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | Register on Page |
|-----------------------|-------------------------------|-----------|----------|--------|-----------------------|----------|---------|--------|------------------|
| ADCON0 | ADFM | VCFG | CHS<3:0> | | | | GO/DONE | ADON | 106 |
| ADCON1 | — | ADCS<2:0> | | | — | — | — | — | 107 |
| ANSELA | — | — | ANSA5 | ANSA4 | — | ANSA2 | ANSA1 | ANSA0 | 50 |
| ADRESH ⁽²⁾ | A/D Result Register High Byte | | | | | | | | 108* |
| ADRESL ⁽²⁾ | A/D Result Register Low Byte | | | | | | | | 106* |
| PORTA | — | — | RA5 | RA4 | RA3 | RA2 | RA1 | RA0 | 48 |
| INTCON | GIE | PEIE | T0IE | INTE | IOCIE | T0IF | INTF | IOCIF | 20 |
| PIE1 | TMR1GIE | ADIE | — | — | — | HLTMR1IE | TMR2IE | TMR1IE | 21 |
| PIR1 | TMR1GIF | ADIF | — | — | — | HLTMR1IF | TMR2IF | TMR1IF | 23 |
| TRISA | — | — | TRISA5 | TRISA4 | TRISA3 ⁽¹⁾ | TRISA2 | TRISA1 | TRISA0 | 48 |

Legend: x = unknown, u = unchanged, — = unimplemented read as '0'. Shaded cells are not used for ADC module.

* Page provides register information.

Note 1: TRISA3 always reads '1'.

2: Read-only register.

PIC12F752/HV752

NOTES:

13.0 FIXED VOLTAGE REFERENCE (FVR)

The Fixed Voltage Reference (FVR) is a stable voltage reference, independent of V_{DD} , with 1.2V output level. The output of the FVR can be configured to supply a reference voltage to the following:

- ADC input channel
- Comparator 1 positive input (C1VP)
- Comparator 2 positive input (C2VP)
- REFOUT pin

On the PIC12F752, the FVR is enabled by setting the FVREN bit of the FVRCON register. The FVR is always enabled on the PIC12HV752 device.

13.1 Fixed Voltage Reference Output

The FVR output can be applied to the REFOUT pin by setting the FVRBUFSS and FVRBUFEN bits of the FVRCON register. The FVRBUFSS bit selects either the FVR or DAC output reference to the REFOUT pin buffer. The FVRBUFEN bit enables the output buffer to the REFOUT pin.

Enabling the REFOUT pin automatically overrides any digital input or output functions of the pin. Reading the REFOUT pin when it has been configured for a reference voltage output will always return a '0'.

13.2 FVR Stabilization Period

When the Fixed Voltage Reference module is enabled, it requires time for the reference circuit to stabilize. Once the circuit stabilizes and is ready for use, the FVRRDY bit of the FVRCON register will be set. See [Section 20.0 "Electrical Specifications"](#) for the minimum delay requirement.

13.3 Operation During Sleep

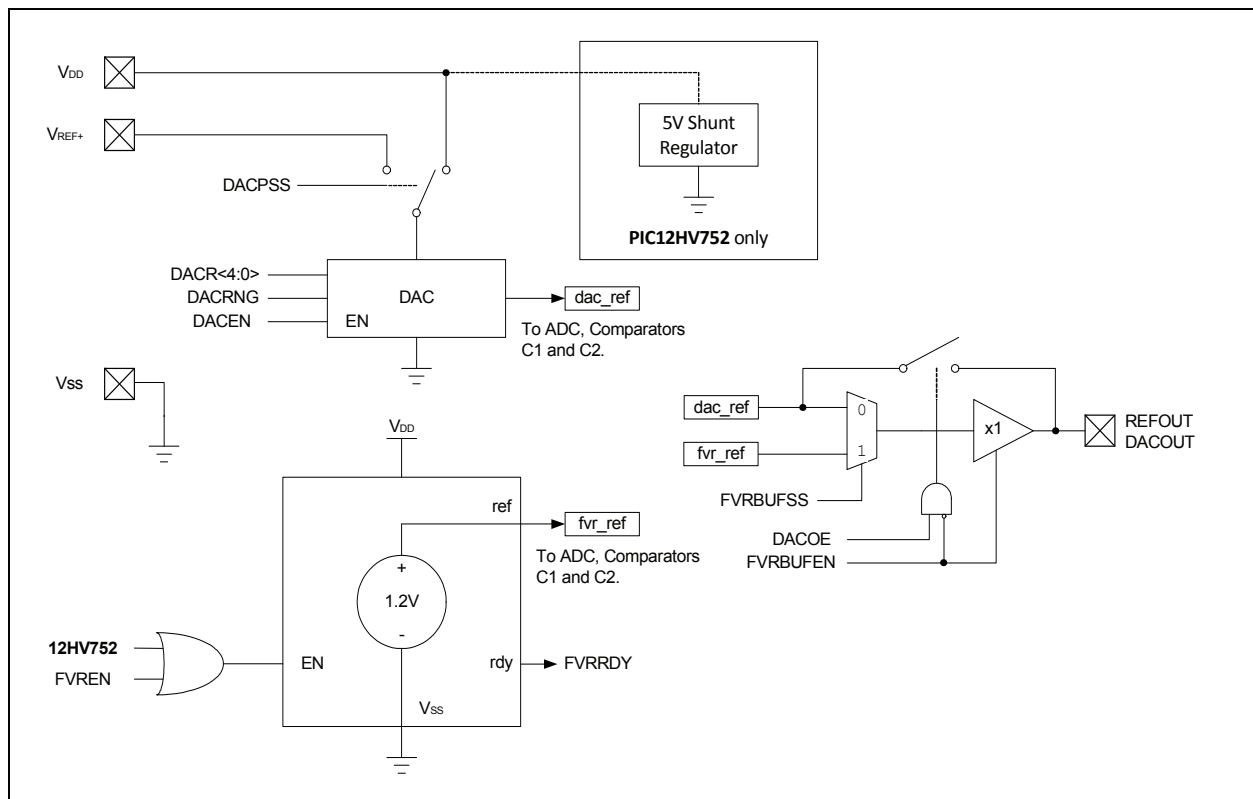
When the device wakes up from Sleep through an interrupt or a Watchdog Timer time-out, the contents of the FVRCON register are not affected. To minimize current consumption in Sleep mode, FVR the voltage reference should be disabled.

13.4 Effects of a Reset

A device Reset clears the FVRCON register. As a result:

- The FVR module is disabled.
- The FVR voltage output is disabled on the REFOUT pin.

FIGURE 13-1: VOLTAGE REFERENCE BLOCK DIAGRAM



PIC12F752/HV752

13.5 FVR Control Registers

REGISTER 13-1: FVRCON: FIXED VOLTAGE REFERENCE CONTROL REGISTER

| | | | | | | | |
|---------|--------|----------|----------|-----|-----|-----|-------|
| R/W-0/0 | R-q/q | R/W-0/0 | R/W-0/0 | U-0 | U-0 | U-0 | U-0 |
| FVREN | FVRRDY | FVRBUFEN | FVRBUFSS | — | — | — | — |
| bit 7 | | | | | | | bit 0 |

Legend:

| | | |
|----------------------|----------------------|---|
| R = Readable bit | W = Writable bit | U = Unimplemented bit, read as '0' |
| u = Bit is unchanged | x = Bit is unknown | -n/n = Value at POR and BOR/Value at all other Resets |
| '1' = Bit is set | '0' = Bit is cleared | q = Value depends on condition |

- bit 7 **FVREN:** Fixed Voltage Reference Enable bit
0 = Fixed Voltage Reference is disabled
1 = Fixed Voltage Reference is enabled
- bit 6 **FVRRDY:** Fixed Voltage Reference Ready Flag bit
0 = Fixed Voltage Reference output is not ready or not enabled bit
1 = Fixed Voltage Reference output is ready for use
- bit 5 **FVRBUFEN:** Voltage Reference Output Pin Buffer Enable
0 = Output buffer is disabled
1 = Output buffer is enabled
- bit 4 **FVRBUFSS:** Voltage Reference Pin Buffer Source Select bit
0 = Bandgap (1.2V) is the input to the output voltage buffer
1 = dac_out is the input to the output voltage buffer
- bit 3-0 **Unimplemented:** Read as '0'

TABLE 13-1: SUMMARY OF REGISTERS ASSOCIATED WITH FIXED VOLTAGE REFERENCE

| Name | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | Register on page |
|--------|-------|--------|----------|----------|-------|-------|-------|-------|------------------|
| FVRCON | FVREN | FVRRDY | FVRBUFEN | FVRBUFSS | — | — | — | — | 114 |

Legend: Shaded cells are not used with the Fixed Voltage Reference.

14.0 DIGITAL-TO-ANALOG CONVERTER (DAC) MODULE

The 5-bit, dual range Digital-to-Analog Converter (DAC) module supplies a variable voltage reference, with 64 selectable output levels of which 3 levels are duplicated. The output is ratiometric with respect to the input source, VSRC+. See Figure 14-1 for a block diagram of the DAC module.

The input of the DAC can be connected to two external voltage connections:

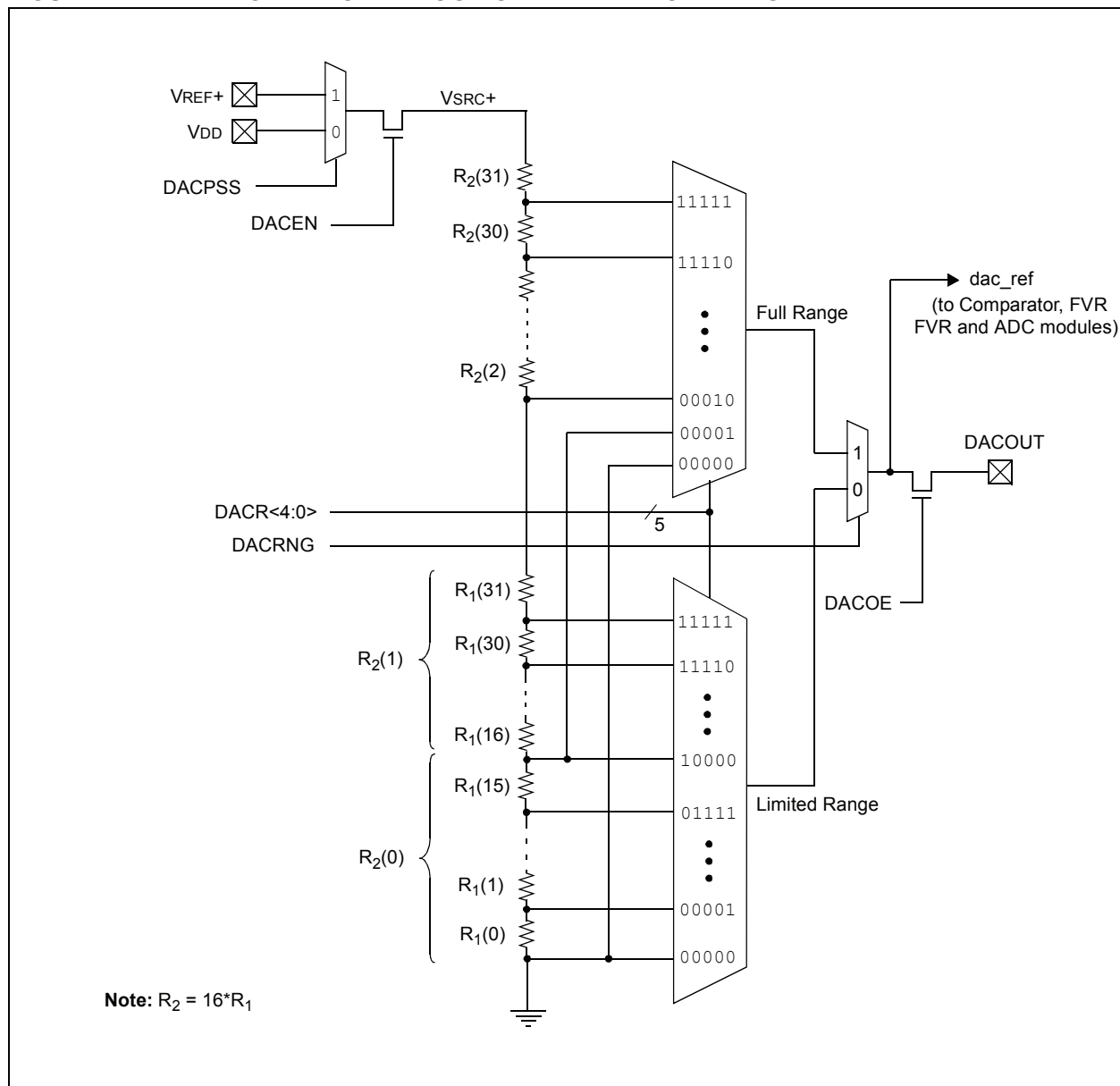
- VDD pin
- VREF+ pin

The output of the DAC module provides a reference voltage to the following:

- Comparator positive input
- ADC input channel
- FVR input reference
- DACOUT pin

The DAC is enabled by setting the DACEN bit of the DACCON0 register.

FIGURE 14-1: DIGITAL-TO-ANALOG CONVERTER BLOCK DIAGRAM



PIC12F752/HV752

14.1 DAC Positive Voltage Source

The DACPSS bit of the DACCON0 register selects the positive voltage source, VSRC+. The following voltage sources are available:

- VDD pin (default)
- VREF+ pin

DAC module can select the positive voltage source using the DACPSS bit of the DACCON0 register. The default source, DACPSS = 0, connects VDD to the positive voltage source (VSRC+). VSRC+ can be changed to the VREF+ pin by setting DACPSS = 1.

14.2 DAC Range Selection

The DACRNG bit of the DACCON0 register selects between full-range or limited-range DAC output voltage.

Each range selects the output in 32 equal steps.

In Full-Range mode, the output is $(31/32) \cdot VSRC+$. In Limited-Range mode, the maximum VOUT is limited to 6% of VSRC+, $(31/512) \cdot VSRC+$.

14.3 Ratiometric Output Level

The DAC output value is derived using a resistor ladder with one end of the ladder tied to the positive voltage reference and the other end tied to VSS. If the voltage of the input source fluctuates, a similar fluctuation will result in the DAC output value.

The resistor values within the ladder can be found in [Section 20.0 “Electrical Specifications”](#).

14.4 DAC Output Voltage

The DAC output voltage level of the DAC is determined by the DACRNG and the DACR<4:0> bits of the DACCON0 and DACCON1 registers, respectively.

Use [Equation 14-1](#) to determine the value of the DAC output voltage. [Example 14-1](#) illustrates the calculations of the minimum, maximum and increment size of the DAC output voltage in Full Range mode. [Example 14-2](#) illustrates the Limited Range mode of the DAC output voltage values.

EQUATION 14-1: DAC OUTPUT VOLTAGE

$$V_{OUT} = \left((VSRC+) \left(\frac{DACR\langle 4:0 \rangle}{2^n} \right) \right)$$

Note: The value of 'n' is determined by the DACRNG bit.
When: DACRNG = 0 (Limited Range mode); n = 9;
DACRNG = 1 (Full Range mode); n = 5.

EXAMPLE 14-1: FULL RANGE MODE

Given: VSRC = VDD = 5V, DACRNG = 1

$$V_{OUT} = [VSRC+ * (DACR\langle 4:0 \rangle / 2^5)]$$

Minimum Vout Calculation:

DACR<4:0> = 0 0000b, (0d);

$$V_{OUT} = [5V * (0/32)] = 0V;$$

Maximum Vout Calculation:

DACR<4:0> = 1 1111b, (31d);

$$V_{OUT} = [5V * (31/32)] = 4.84V;$$

Step Increment Calculation:

DACR<4:0> = 0 0001b, (1d);

$$V_{OUT} = [5V * (1/32)] = 156 \text{ mV}$$

Full Range Mode Operation:

0V ≤ VOUT ≤ 4.84V, with 32-step increments of 156 mV.

EXAMPLE 14-2: LIMITED RANGE MODE

Given: VSRC = VDD = 5V, DACRNG = 0

$$V_{OUT} = [VSRC+ * (DACR\langle 4:0 \rangle / 2^9)]$$

Minimum Vout Calculation:

DACR<4:0> = 0 0000b, (0d);

$$V_{OUT} = [5V * (0/512)] = 0V;$$

Maximum Vout Calculation:

DACR<4:0> = 1 1111b, (31d);

$$V_{OUT} = [5V * (31/512)] = 303 \text{ mV};$$

Step Increment Calculation:

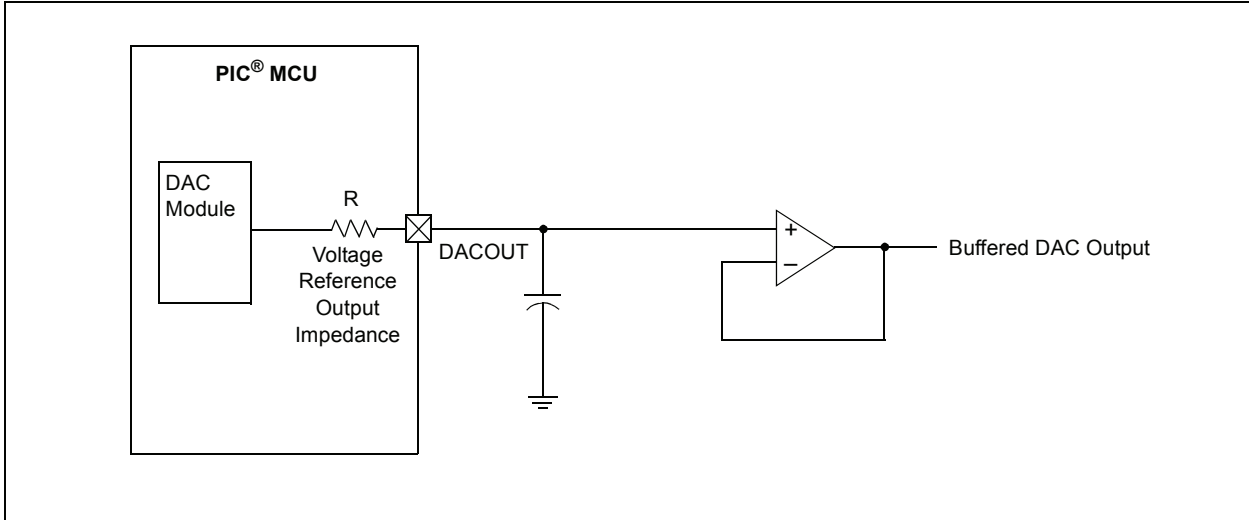
DACR<4:0> = 0 0001b, (1d);

$$V_{OUT} = [5V * (1/512)] = 9.8 \text{ mV}$$

Limited Range Mode Operation:

0V ≤ VOUT ≤ 303 mV, with 32-step increments of 9.8 mV.

FIGURE 14-2: VOLTAGE REFERENCE OUTPUT BUFFER EXAMPLE



14.5 DAC Voltage Reference Output

The DAC output (*dac_ref*) can be applied to the DACOUT pin as an unbuffered signal by:

- Setting the DACOE bit of the DACCON0 register
- Clearing the FVRBUFSS bit of the FVRCON register.
- Clearing the FVRBUFEN bit of the FVRCON register.

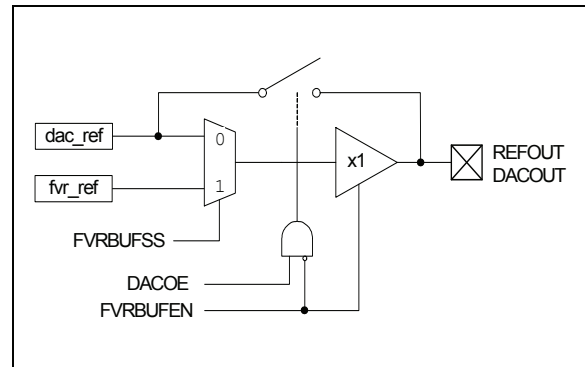
Figure 14-3 shows a block diagram pin configuration for the *dac_ref* and *fvr_ref* signals. This unbuffered DACOUT pin has limited current drive capability. When a higher drive current is required, an external buffer can be used on the DACOUT pin. Figure 14-2 shows an example of buffering technique.

The DAC output can also be configured to use an internal buffer by:

- Setting the FVRBUFEN bit of the FVRCON register changing the pin configuration to be the REFOUT pin.

Enabling the DACOUT pin automatically overrides any digital input or output functions of the pin. Reading the DACOUT pin when it has been configured for DAC reference voltage output will always return a '0'.

FIGURE 14-3: DAC/FVR OUTPUT PIN



14.6 Operation During Sleep

When the device wakes up from Sleep through an interrupt or a Watchdog Timer time-out, the contents of the DACCON0 register are not affected. To minimize current consumption in Sleep mode, the voltage reference should be disabled.

14.7 Effects of a Reset

A device Reset clears the DACCON0 and DACCON1 registers. As a result:

- the DAC module is disabled.
- the DAC voltage output is disabled on the DACOUT pin.

PIC12F752/HV752

14.8 DAC Control Registers

REGISTER 14-1: DACCON0: VOLTAGE REFERENCE CONTROL REGISTER 0

| | | | | | | | |
|---------|---------|---------|-----|-----|---------|-----|-------|
| R/W-0/0 | R/W-0/0 | R/W-0/0 | U-0 | U-0 | R/W-0/0 | U-0 | U-0 |
| DACEN | DACRNG | DACOE | — | — | DACPSS | — | — |
| bit 7 | | | | | | | bit 0 |

Legend:

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

bit 7 **DACEN:** DAC Enable bit
1 = DAC is enabled
0 = DAC is disabled

bit 6 **DACRNG:** DAC Range Selection bit⁽¹⁾
1 = DAC is operating in Full Range mode
0 = DAC is operating in Limited Range mode

bit 5 **DACOE:** DAC Voltage Output Enable bit
1 = DAC reference output is enabled to the DACOUT pin⁽²⁾
0 = DAC reference output is disabled

bit 4-3 **Unimplemented:** Read as '0'

bit 2 **DACPSS:** DAC Positive Source Select bits
0 = VDD
1 = VREF+ pin

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

Note 1: Refer to [Equation 14-1](#).

2: The DACOUT pin configuration requires additional control bits in the FVRCON register (see [Figure 14-3](#)).

REGISTER 14-2: DACCON1: VOLTAGE REFERENCE CONTROL REGISTER 1

| | | | | | | | |
|-------|-----|-----|-----------|---------|---------|---------|---------|
| U-0 | U-0 | U-0 | R/W-0/0 | R/W-0/0 | R/W-0/0 | R/W-0/0 | R/W-0/0 |
| — | — | — | DACR<4:0> | | | | — |
| bit 7 | | | | | | | bit 0 |

Legend:

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

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

bit 4-0 **DACR<4:0>:** DAC Voltage Output Select bits
1 1111 = DAC Voltage Maximum Output
•
•
•
0 0000 = DAC Voltage Minimum Output

Note 1: Refer to [Equation 14-1](#) to calculate the value of the DAC Voltage Output.

TABLE 14-1: SUMMARY OF REGISTERS ASSOCIATED WITH THE DAC MODULE

| Name | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | Register on page |
|---------|-------|--------|----------|-----------|-------|--------|-------|-------|------------------|
| DACCON0 | DACEN | DACRNG | DACOE | — | — | DACPSS | — | — | 118 |
| DACCON1 | — | — | — | DACR<4:0> | | | | | 118 |
| FVRCON | FVREN | FVRRDY | FVRBUFEN | FVRBUFSS | — | — | — | — | 114 |

Legend: — = unimplemented, read as '0'. Shaded cells are unused by the DAC module.

PIC12F752/HV752

NOTES:

15.0 COMPARATOR MODULE

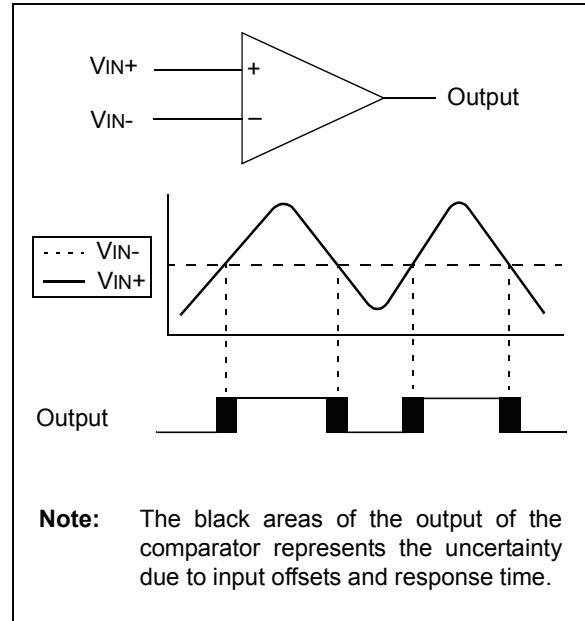
Comparators are used to interface analog circuits to a digital circuit by comparing two analog voltages and providing a digital indication of their relative magnitudes. Comparators are very useful mixed signal building blocks because they provide analog functionality independent of program execution. The analog comparator module includes the following features:

- Independent comparator control
- Programmable input selection
- Comparator output is available internally/externally
- Programmable output polarity
- Interrupt-on-change
- Wake-up from Sleep
- Programmable Speed/Power optimization
- PWM shutdown
- Programmable and fixed voltage reference

15.1 Comparator Overview

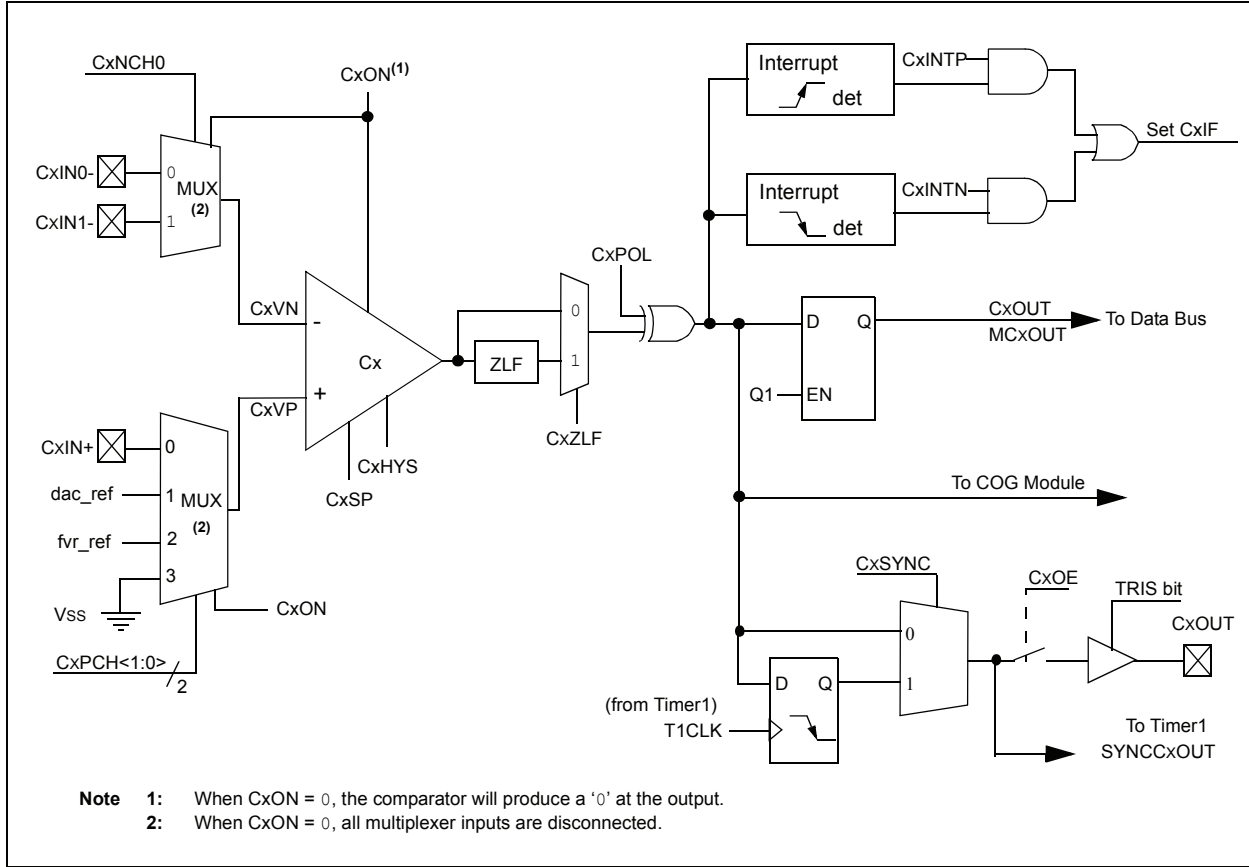
A single comparator is shown in [Figure 15-1](#) along with the relationship between the analog input levels and the digital output. When the analog voltage at V_{IN+} is less than the analog voltage at V_{IN-} , the output of the comparator is a digital low level. When the analog voltage at V_{IN+} is greater than the analog voltage at V_{IN-} , the output of the comparator is a digital high level.

FIGURE 15-1: SINGLE COMPARATOR



PIC12F752/HV752

FIGURE 15-2: COMPARATOR MODULE SIMPLIFIED BLOCK DIAGRAM



15.2 Comparator Control

Each comparator has 2 control registers: CMxCON0 and CMxCON1.

The CMxCON0 registers (see [Register 15-1](#)) contain Control and Status bits for the following:

- Enable
- Output selection
- Output pin enable
- Output polarity
- Speed/Power selection
- Hysteresis enable
- Output synchronization

The CMxCON1 registers (see [Register 15-2](#)) contain Control bits for the following:

- Interrupt edge polarity (rising and/or falling)
- Positive input channel selection
- Negative input channel selection

15.2.1 COMPARATOR ENABLE

Setting the CxON bit of the CMxCON0 register enables the comparator for operation. Clearing the CxON bit disables the comparator resulting in minimum current consumption.

15.2.2 COMPARATOR OUTPUT SELECTION

The output of the comparator can be monitored by reading either the CxOUT bit of the CMxCON0 register or the MCxOUT bit of the CMOUT register. In order to make the output available for an external connection, the following conditions must be true:

- CxOE bit of the CMxCON0 register must be set
- Corresponding TRIS bit must be cleared
- CxON bit of the CMxCON0 register must be set

Note 1: The CxOE bit of the CMxCON0 register overrides the PORT data latch. Setting the CxON bit of the CMxCON0 register has no impact on the port override.

2: The internal output of the comparator is latched with each instruction cycle. Unless otherwise specified, external outputs are not latched.

15.2.3 COMPARATOR OUTPUT POLARITY

Inverting the output of the comparator is functionally equivalent to swapping the comparator inputs. The polarity of the comparator output can be inverted by setting the CxPOL bit of the CMxCON0 register. Clearing the CxPOL bit results in a non-inverted output.

[Table 15-1](#) shows the output state versus input conditions, including polarity control.

TABLE 15-1: COMPARATOR OUTPUT STATE VS. INPUT CONDITIONS

| Input Condition | CxPOL | CxOUT |
|-----------------|-------|-------|
| $CxVN > CxVP$ | 0 | 0 |
| $CxVN < CxVP$ | 0 | 1 |
| $CxVN > CxVP$ | 1 | 1 |
| $CxVN < CxVP$ | 1 | 0 |

15.2.4 COMPARATOR SPEED/POWER SELECTION

The trade-off between speed or power can be optimized during program execution with the CxSP control bit. The default state for this bit is '1' which selects the normal speed mode. Device power consumption can be optimized at the cost of slower comparator propagation delay by clearing the CxSP bit to '0'.

15.3 Comparator Hysteresis

A selectable amount of separation voltage can be added to the input pins of each comparator to provide a hysteresis function to the overall operation. Hysteresis is enabled by setting the CxHYS bit of the CMxCON0 register.

See [Section 20.0 "Electrical Specifications"](#) for more information.

15.4 Timer1 Gate Operation

The output resulting from a comparator operation can be used as a source for gate control of Timer1. See [Section 7.5 "Timer1 Gate"](#) for more information. This feature is useful for timing the duration or interval of an analog event.

It is recommended that the comparator output be synchronized to Timer1. This ensures that Timer1 does not increment while a change in the comparator is occurring.

PIC12F752/HV752

15.4.1 COMPARATOR OUTPUT SYNCHRONIZATION

The output from either comparator, C1 or C2, can be synchronized with Timer1 by setting the CxSYNC bit of the CMxCON0 register.

Once enabled, the comparator output is latched on the falling edge of the Timer1 source clock. If a prescaler is used with Timer1, the comparator output is latched after the prescaling function. To prevent a race condition, the comparator output is latched on the falling edge of the Timer1 clock source and Timer1 increments on the rising edge of its clock source. See the Comparator Block Diagram (Figure 15-2) and the Timer1 Block Diagram (Figure 7-1) for more information.

15.5 Comparator Interrupt

An interrupt can be generated upon a change in the output value of the comparator for each comparator, a rising edge detector and a Falling edge detector are present.

When either edge detector is triggered and its associated enable bit is set (CxINTP and/or CxINTN bits of the CMxCON1 register), the Corresponding Interrupt Flag bit (CxIF bit of the PIR2 register) will be set.

To enable the interrupt, you must set the following bits:

- CxON, CxPOL and CxSP bits of the CMxCON0 register
- CxIE bit of the PIE2 register
- CxINTP bit of the CMxCON1 register (for a rising edge detection)
- CxINTN bit of the CMxCON1 register (for a falling edge detection)
- PEIE and GIE bits of the INTCON register

The associated interrupt flag bit, CxIF bit of the PIR2 register, must be cleared in software. If another edge is detected while this flag is being cleared, the flag will still be set at the end of the sequence.

Note: Although a comparator is disabled, an interrupt can be generated by changing the output polarity with the CxPOL bit of the CMxCON0 register, or by switching the comparator on or off with the CxON bit of the CMxCON0 register.

15.6 Comparator Positive Input Selection

Configuring the CxPCH<1:0> bits of the CMxCON1 register directs an internal voltage reference or an analog pin to the non-inverting input of the comparator:

- CxIN0+ analog pin
- DAC Reference Voltage (dac_ref)
- FVR Reference Voltage (fvr_ref)
- Vss (Ground)

See [Section 13.0 “Fixed Voltage Reference \(FVR\)”](#) for more information on the Fixed Voltage Reference module.

See [Section 14.0 “Digital-to-Analog Converter \(DAC\) Module”](#) for more information on the DAC input signal.

Any time the comparator is disabled (CxON = 0), all comparator inputs are disabled.

15.7 Comparator Negative Input Selection

The CxNCH0 bit of the CMxCON0 register selects the analog input pin to the comparator inverting input.

Note: To use CxIN0+ and CxIN1x- pins as analog input, the appropriate bits must be set in the ANSEL register and the corresponding TRIS bits must also be set to disable the output drivers.

15.8 Comparator Response Time

The comparator output is indeterminate for a period of time after the change of an input source or the selection of a new reference voltage. This period is referred to as the response time. The response time of the comparator differs from the settling time of the voltage reference. Therefore, both of these times must be considered when determining the total response time to a comparator input change. See the Comparator and Voltage Reference Specifications in [Section 20.0 “Electrical Specifications”](#) for more details.

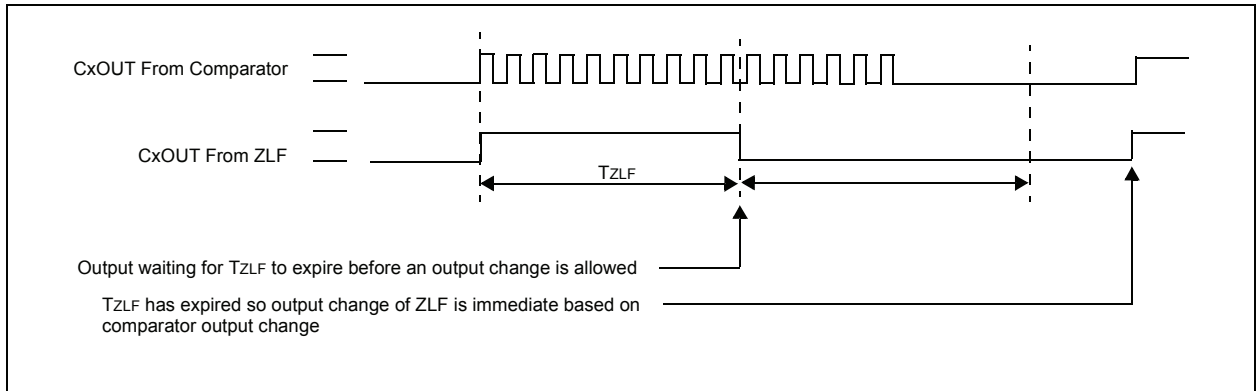
15.9 Interaction with the COG Module

The comparator outputs can be brought to the COG module in order to facilitate auto-shutdown. If auto-restart is also enabled, the comparators can be configured as a closed loop analog feedback to the COG, thereby creating an analog controlled PWM.

15.10 Zero Latency Filter

In high-speed operation, and under proper circuit conditions, it is possible for the comparator output to oscillate. This oscillation can have adverse effects on the hardware and software relying on this signal. Therefore, a digital filter has been added to the comparator output to suppress the comparator output oscillation. Once the comparator output changes, the output is prevented from reversing the change for a nominal time of 20 ns. This allows the comparator output to stabilize without affecting other dependent devices. Refer to [Figure 15-3](#).

FIGURE 15-3: COMPARATOR ZERO LATENCY FILTER OPERATION



PIC12F752/HV752

15.11 Analog Input Connection Considerations

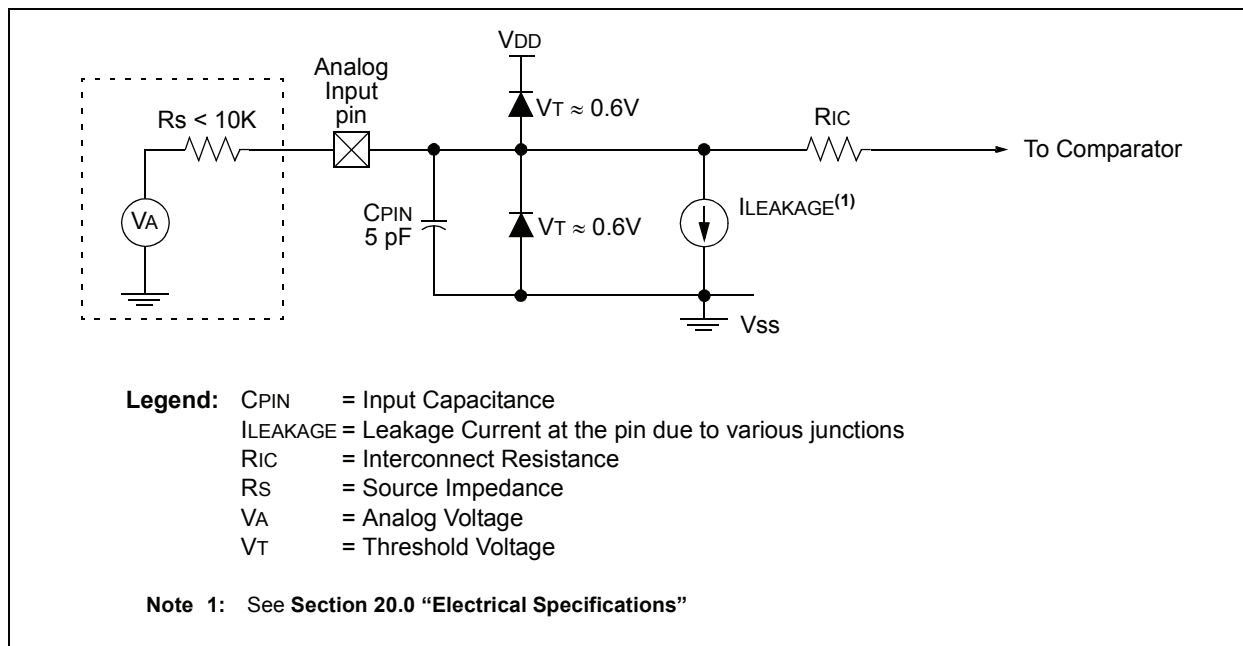
A simplified circuit for an analog input is shown in Figure 15-4. Since the analog input pins share their connection with a digital input, they have reverse biased ESD protection diodes to V_{DD} and V_{SS} . The analog input, therefore, must be between V_{SS} and V_{DD} . If the input voltage deviates from this range by more than 0.6V in either direction, one of the diodes is forward biased and a latch-up may occur.

A maximum source impedance of 10 k Ω is recommended for the analog sources. Also, any external component connected to an analog input pin, such as a capacitor or a Zener diode, should have very little leakage current to minimize inaccuracies introduced.

Note 1: When reading a PORT register, all pins configured as analog inputs will read as a '0'. Pins configured as digital inputs will convert as an analog input, according to the input specification.

2: Analog levels on any pin defined as a digital input, may cause the input buffer to consume more current than is specified.

FIGURE 15-4: ANALOG INPUT MODEL



15.12 Comparator Control Registers

REGISTER 15-1: CMxCON0: COMPARATOR Cx CONTROL REGISTER 0

| | | | | | | | |
|---------|-------|---------|---------|---------|---------|---------|---------|
| R/W-0/0 | R-0/0 | R/W-0/0 | R/W-0/0 | R/W-0/0 | R/W-1/1 | R/W-0/0 | R/W-0/0 |
| CxON | CxOUT | CxOE | CxPOL | CxZLF | CxSP | CxHYS | CxSYNC |
| bit 7 | | | | | | | bit 0 |

Legend:

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

- bit 7 **CxON:** Comparator Enable bit
 1 = Comparator is enabled
 0 = Comparator is disabled and consumes no active power
- bit 6 **CxOUT:** Comparator Output bit
 If CxPOL = 1 (inverted polarity):
 1 = CxVP < CxVN
 0 = CxVP > CxVN
 If CxPOL = 0 (non-inverted polarity):
 1 = CxVP > CxVN
 0 = CxVP < CxVN
- bit 5 **CxOE:** Comparator Output Enable bit
 1 = CxOUT is present on the CxOUT pin. Requires that the associated TRIS bit be cleared to actually drive the pin. Not affected by CxON.
 0 = CxOUT is internal only
- bit 4 **CxPOL:** Comparator Output Polarity Select bit
 1 = Comparator output is inverted
 0 = Comparator output is not inverted
- bit 3 **CxZLF:** Zero Latency Filter Enable bit
 1 = Zero latency filter is enabled
 0 = Zero latency filter is disabled
- bit 2 **CxSP:** Comparator Speed/Power Select bit
 1 = Comparator operates in normal power, higher speed mode
 0 = Comparator operates in low-power, low-speed mode
- bit 1 **CxHYS:** Comparator Hysteresis Enable bit
 1 = Comparator hysteresis enabled
 0 = Comparator hysteresis disabled
- bit 0 **CxSYNC:** Comparator Output Synchronous Mode bit
 1 = Comparator output to Timer1 and I/O pin is synchronous to changes on Timer1 clock source. Output updated on the falling edge of Timer1 clock source.
 0 = Comparator output to Timer1 and I/O pin is asynchronous.

PIC12F752/HV752

REGISTER 15-2: CMxCON1: COMPARATOR Cx CONTROL REGISTER 1

| | | | | | | | |
|---------|---------|------------|---------|-----|-----|-----|---------|
| R/W-0/0 | R/W-0/0 | R/W-0/0 | R/W-0/0 | U-0 | U-0 | U-0 | R/W-0/0 |
| CxINTP | CxINTN | CxPCH<1:0> | | — | — | — | CxNCH0 |
| bit 7 | | | | | | | bit 0 |

Legend:

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

- bit 7 **CxINTP:** Comparator Interrupt on Positive Going Edge Enable bit
1 = The CxIF interrupt flag will be set upon a positive going edge of the CxOUT bit
0 = No interrupt flag will be set on a positive going edge of the CxOUT bit
- bit 6 **CxINTN:** Comparator Interrupt on Negative Going Edge Enable bits
1 = The CxIF interrupt flag will be set upon a negative going edge of the CxOUT bit
0 = No interrupt flag will be set on a negative going edge of the CxOUT bit
- bit 5-4 **CxPCH<1:0>:** Comparator Positive Input Channel Select bits
00 = CxVP connects to CxIN+ pin
01 = CxVP connects to DAC Voltage Reference (dac_ref)
10 = CxVP connects to FVR Voltage Reference (fvr_ref)
11 = CxVP connects to Vss
- bit 3-1 **Unimplemented:** Read as '0'
- bit 0 **CxNCH0:** Comparator Negative Input Channel Select bits
0 = CxVN connects to CxIN0- pin
1 = CxVN connects to CxIN1- pin

REGISTER 15-3: CMOUT: COMPARATOR OUTPUT REGISTER

| | | | | | | | |
|-------|-----|-----|-----|-----|-----|--------|--------|
| U-0 | U-0 | U-0 | U-0 | U-0 | U-0 | R-0/0 | R-0/0 |
| — | — | — | — | — | — | MC2OUT | MC1OUT |
| bit 7 | | | | | | | bit 0 |

Legend:

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

- bit 7-2 **Unimplemented:** Read as '0'
- bit 1 **MC2OUT:** Mirror Copy of C2OUT bit
- bit 0 **MC1OUT:** Mirror Copy of C1OUT bit

TABLE 15-2: SUMMARY OF REGISTERS ASSOCIATED WITH COMPARATOR MODULE

| Name | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | Register on Page |
|---------|--------|--------|---------------|---------------|-----------------------|---------|--------|--------|------------------|
| CM1CON0 | C1ON | C1OUT | C1OE | C1POL | C1ZLF | C1SP | C1HYS | C1SYNC | 127 |
| CM1CON1 | C1INTP | C1INTN | C1PCH<1:0> | | — | — | — | C1NCH0 | 128 |
| CM2CON0 | C2ON | C2OUT | C2OE | C2POL | C2ZLF | C2SP | C2HYS | C2SYNC | 127 |
| CM2CON1 | C2INTP | C2INTN | C2PCH<1:0> | | — | — | — | C2NCH0 | 128 |
| CMOUT | — | — | — | — | — | — | MCOUT2 | MCOUT1 | 128 |
| DACCON0 | DACEN | DACRNG | DACOE | — | — | DACPSS0 | — | — | 118 |
| DACCON1 | — | — | — | DACR<4:0> | | | | — | 118 |
| FVRCON | FVREN | FVRRDY | FVR- BUFEN | FVR- BUFSS | — | — | — | — | 114 |
| INTCON | GIE | PEIE | T0IE | INTE | IOCIE | T0IF | INTF | IOCIF | 20 |
| PIE2 | — | — | C2IE | C1IE | — | COG1IE | — | CCP1IE | 22 |
| PIR2 | — | — | C2IF | C1IF | — | COG1IF | — | CCP1IF | 24 |
| TRISA | — | — | TRISA5 | TRISA4 | TRISA3 ⁽¹⁾ | TRISA2 | TRISA1 | TRISA0 | 48 |
| ANSELA | — | — | ANSA5 | ANSA4 | — | ANSA2 | ANSA1 | ANSA0 | 50 |

Legend: — = unimplemented location, read as '0'. Shaded cells are unused by the comparator module.

Note 1: TRISA3 always reads '1'.

PIC12F752/HV752

NOTES:

16.0 INSTRUCTION SET SUMMARY

The PIC12F752/HV752 instruction set is highly orthogonal and is comprised of three basic categories:

- **Byte-oriented** operations
- **Bit-oriented** operations
- **Literal and control** operations

Each PIC16 instruction is a 14-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 formats for each of the categories is presented in Figure 16-1, while the various opcode fields are summarized in Table 16-1.

Table 16-2 lists the instructions recognized by the MPASM™ assembler.

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

The destination designator specifies where the result of the operation is to be placed. If 'd' is zero, the result is placed in the W register. If 'd' is one, 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 bit affected by the operation, while 'f' represents the address of the file in which the bit is located.

For **literal and control** operations, 'k' represents an 8-bit or 11-bit constant, or literal value.

One instruction cycle consists of four oscillator periods; for an oscillator frequency of 4 MHz, this gives a normal instruction execution time of 1 μs. 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. When this occurs, the execution takes two instruction cycles, with the second cycle executed as a **NOOP**.

All instruction examples use the format '0xhh' to represent a hexadecimal number, where 'h' signifies a hexadecimal digit.

16.1 Read-Modify-Write Operations

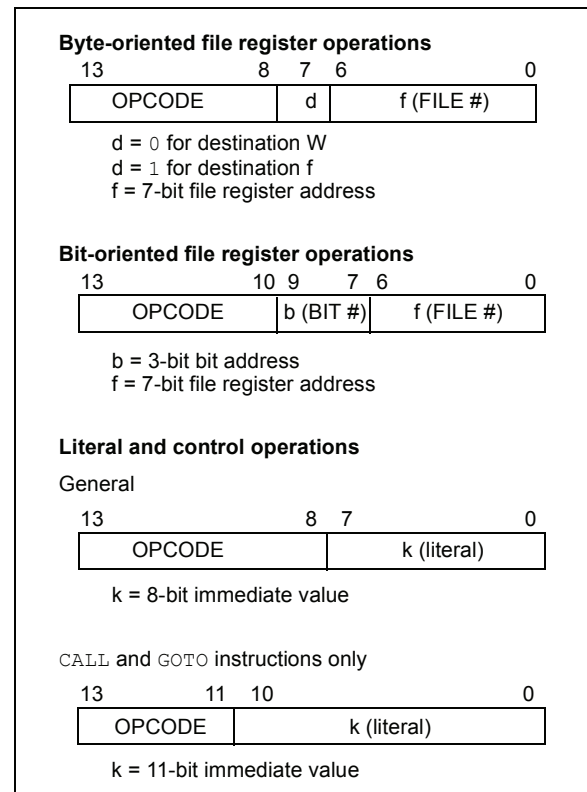
Any instruction that specifies a file register as part of the instruction performs a Read-Modify-Write (RMW) operation. The register is read, the data is modified, and the result is stored according to either the instruction or the destination designator 'd'. A read operation is performed on a register even if the instruction writes to that register.

For example, a `CLRF PORTA` instruction will read PORTA, clear all the data bits, then write the result back to PORTA. This example would have the unintended consequence of clearing the condition that set the IOCIF flag.

TABLE 16-1: 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). 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; d = 0: store result in W, d = 1: store result in file register f. Default is d = 1. |
| PC | Program Counter |
| \overline{TO} | Time-out bit |
| C | Carry bit |
| DC | Digit carry bit |
| Z | Zero bit |
| \overline{PD} | Power-down bit |

FIGURE 16-1: GENERAL FORMAT FOR INSTRUCTIONS



PIC12F752/HV752

TABLE 16-2: PIC12F752/HV752 INSTRUCTION SET

| Mnemonic, Operands | Description | Cycles | 14-Bit Opcode | | Status Affected | Notes | |
|---|-------------|------------------------------|---------------|-----|--------------------|--------------------------------|---------|
| | | | MSb | LSb | | | |
| BYTE-ORIENTED FILE REGISTER OPERATIONS | | | | | | | |
| ADDWF | f, d | Add W and f | 1 | 00 | 0111 dfff ffff | C, DC, Z | 1, 2 |
| ANDWF | f, d | AND W with f | 1 | 00 | 0101 dfff ffff | Z | 1, 2 |
| CLRF | f | Clear f | 1 | 00 | 0001 1fff ffff | Z | 2 |
| CLRWF | – | Clear W | 1 | 00 | 0001 0xxx xxxx | Z | |
| COMF | f, d | Complement f | 1 | 00 | 1001 dfff ffff | Z | 1, 2 |
| DECf | f, d | Decrement f | 1 | 00 | 0011 dfff ffff | Z | 1, 2 |
| DECFSZ | f, d | Decrement f, Skip if 0 | 1(2) | 00 | 1011 dfff ffff | | 1, 2, 3 |
| INCF | f, d | Increment f | 1 | 00 | 1010 dfff ffff | Z | 1, 2 |
| INCFSZ | f, d | Increment f, Skip if 0 | 1(2) | 00 | 1111 dfff ffff | | 1, 2, 3 |
| IORWF | f, d | Inclusive OR W with f | 1 | 00 | 0100 dfff ffff | Z | 1, 2 |
| MOVF | f, d | Move f | 1 | 00 | 1000 dfff ffff | Z | 1, 2 |
| MOVWF | f | Move W to f | 1 | 00 | 0000 1fff ffff | | |
| NOP | – | No Operation | 1 | 00 | 0000 0xx0 0000 | | |
| RLF | f, d | Rotate Left f through Carry | 1 | 00 | 1101 dfff ffff | C | 1, 2 |
| RRF | f, d | Rotate Right f through Carry | 1 | 00 | 1100 dfff ffff | C | 1, 2 |
| SUBWF | f, d | Subtract W from f | 1 | 00 | 0010 dfff ffff | C, DC, Z | 1, 2 |
| SWAPF | f, d | Swap nibbles in f | 1 | 00 | 1110 dfff ffff | | 1, 2 |
| XORWF | f, d | Exclusive OR W with f | 1 | 00 | 0110 dfff ffff | Z | 1, 2 |
| BIT-ORIENTED FILE REGISTER OPERATIONS | | | | | | | |
| BCF | f, b | Bit Clear f | 1 | 01 | 00bb bfff ffff | | 1, 2 |
| BSF | f, b | Bit Set f | 1 | 01 | 01bb bfff ffff | | 1, 2 |
| BTFSC | f, b | Bit Test f, Skip if Clear | 1(2) | 01 | 10bb bfff ffff | | 3 |
| BTFSS | f, b | Bit Test f, Skip if Set | 1(2) | 01 | 11bb bfff ffff | | 3 |
| LITERAL AND CONTROL OPERATIONS | | | | | | | |
| ADDLW | k | Add literal and W | 1 | 11 | 111x kkkk kkkk | C, DC, Z | |
| ANDLW | k | AND literal with W | 1 | 11 | 1001 kkkk kkkk | Z | |
| CALL | k | Call Subroutine | 2 | 10 | 0kkk kkkk kkkk | | |
| CLRWD _T | – | Clear Watchdog Timer | 1 | 00 | 0000 0110 0100 | $\overline{TO}, \overline{PD}$ | |
| GOTO | k | Go to address | 2 | 10 | 1kkk kkkk kkkk | | |
| IORLW | k | Inclusive OR literal with W | 1 | 11 | 1000 kkkk kkkk | Z | |
| MOVLW | k | Move literal to W | 1 | 11 | 00xx kkkk kkkk | | |
| RETFIE | – | Return from interrupt | 2 | 00 | 0000 0000 1001 | | |
| RETLW | k | Return with literal in W | 2 | 11 | 01xx kkkk kkkk | | |
| RETURN | – | Return from Subroutine | 2 | 00 | 0000 0000 1000 | | |
| SLEEP | – | Go into Standby mode | 1 | 00 | 0000 0110 0011 | $\overline{TO}, \overline{PD}$ | |
| SUBLW | k | Subtract W from literal | 1 | 11 | 110x kkkk kkkk | C, DC, Z | |
| XORLW | k | Exclusive OR literal with W | 1 | 11 | 1010 kkkk kkkk | Z | |

- Note 1:** When an I/O register is modified as a function of itself (e.g., `MOVWF PORTA, 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'.
- 2:** If this instruction is executed on the TMR0 register (and where applicable, d = 1), the prescaler will be cleared if assigned to the Timer0 module.
- 3:** If the Program Counter (PC) is modified, or a conditional test is true, the instruction requires two cycles. The second cycle is executed as a `NOP`.

16.2 Instruction Descriptions

ADDLW **Add literal and W**

Syntax: `[label] ADDLW k`

Operands: $0 \leq k \leq 255$

Operation: $(W) + k \rightarrow (W)$

Status Affected: C, DC, Z

Description: The contents of the W register are added to the eight-bit literal 'k' and the result is placed in the W register.

BCF **Bit Clear f**

Syntax: `[label] BCF f,b`

Operands: $0 \leq f \leq 127$
 $0 \leq b \leq 7$

Operation: $0 \rightarrow (f)$

Status Affected: None

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

ADDWF **Add W and f**

Syntax: `[label] ADDWF f,d`

Operands: $0 \leq f \leq 127$
 $d \in [0,1]$

Operation: $(W) + (f) \rightarrow (\text{destination})$

Status Affected: C, DC, Z

Description: Add 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'.

BSF **Bit Set f**

Syntax: `[label] BSF f,b`

Operands: $0 \leq f \leq 127$
 $0 \leq b \leq 7$

Operation: $1 \rightarrow (f)$

Status Affected: None

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

ANDLW **AND literal with W**

Syntax: `[label] ANDLW k`

Operands: $0 \leq k \leq 255$

Operation: $(W) .\text{AND.} (k) \rightarrow (W)$

Status Affected: Z

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

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

Syntax: `[label] BTFSC f,b`

Operands: $0 \leq f \leq 127$
 $0 \leq b \leq 7$

Operation: skip if $(f) = 0$

Status Affected: None

Description: If bit 'b' in register 'f' is '1', the next instruction is executed.
If bit 'b' in register 'f' is '0', the next instruction is discarded, and a NOP is executed instead, making this a two-cycle instruction.

ANDWF **AND W with f**

Syntax: `[label] ANDWF f,d`

Operands: $0 \leq f \leq 127$
 $d \in [0,1]$

Operation: $(W) .\text{AND.} (f) \rightarrow (\text{destination})$

Status Affected: Z

Description: AND 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'.

PIC12F752/HV752

BTFS Bit Test f, Skip if Set

Syntax: [*label*] BTFS f,b
Operands: $0 \leq f \leq 127$
 $0 \leq b < 7$
Operation: skip if (f) = 1
Status Affected: None
Description: If bit 'b' in register 'f' is '0', the next instruction is executed.
If bit 'b' is '1', then the next instruction is discarded and a NOP is executed instead, making this a two-cycle instruction.

CLRWD Clear Watchdog Timer

Syntax: [*label*] CLRWD
Operands: None
Operation: 00h → WDT
0 → WDT prescaler,
1 → \overline{TO}
1 → \overline{PD}
Status Affected: \overline{TO} , \overline{PD}
Description: CLRWD instruction resets the Watchdog Timer. It also resets the prescaler of the WDT.
Status bits \overline{TO} and \overline{PD} are set.

CALL Call Subroutine

Syntax: [*label*] CALL k
Operands: $0 \leq k \leq 2047$
Operation: (PC)+ 1 → TOS,
k → PC<10:0>,
(PCLATH<4:3>) → PC<12:11>
Status Affected: None
Description: Call Subroutine. First, return address (PC + 1) is pushed onto the stack. The eleven-bit immediate address is loaded into PC bits <10:0>. The upper bits of the PC are loaded from PCLATH.
CALL is a two-cycle instruction.

COMF Complement f

Syntax: [*label*] COMF f,d
Operands: $0 \leq f \leq 127$
d ∈ [0,1]
Operation: (\bar{f}) → (destination)
Status Affected: Z
Description: The contents of register 'f' are complemented. If 'd' is '0', the result is stored in W. If 'd' is '1', the result is stored back in register 'f'.

CLRF Clear f

Syntax: [*label*] CLRF f
Operands: $0 \leq f \leq 127$
Operation: 00h → (f)
1 → Z
Status Affected: Z
Description: The contents of register 'f' are cleared and the Z bit is set.

DECF Decrement f

Syntax: [*label*] DECF f,d
Operands: $0 \leq f \leq 127$
d ∈ [0,1]
Operation: (f) - 1 → (destination)
Status Affected: Z
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'.

CLRW Clear W

Syntax: [*label*] CLRW
Operands: None
Operation: 00h → (W)
1 → Z
Status Affected: Z
Description: W register is cleared. Zero bit (Z) is set.

DECFSZ Decrement f, Skip if 0

Syntax: [*label*] DECFSZ f,d

Operands: $0 \leq f \leq 127$
 $d \in [0,1]$

Operation: $(f) - 1 \rightarrow (\text{destination})$;
 skip if result = 0

Status Affected: None

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 '1', the next instruction is executed. If the result is '0', then a NOP is executed instead, making it a two-cycle instruction.

INCFSZ Increment f, Skip if 0

Syntax: [*label*] INCFSZ f,d

Operands: $0 \leq f \leq 127$
 $d \in [0,1]$

Operation: $(f) + 1 \rightarrow (\text{destination})$;
 skip if result = 0

Status Affected: None

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 '1', the next instruction is executed. If the result is '0', a NOP is executed instead, making it a two-cycle instruction.

GOTO Unconditional Branch

Syntax: [*label*] GOTO k

Operands: $0 \leq k \leq 2047$

Operation: $k \rightarrow \text{PC}\langle 10:0 \rangle$
 $\text{PCLATH}\langle 4:3 \rangle \rightarrow \text{PC}\langle 12:11 \rangle$

Status Affected: None

Description: GOTO is an unconditional branch. The eleven-bit immediate value is loaded into PC bits $\langle 10:0 \rangle$. The upper bits of PC are loaded from PCLATH $\langle 4:3 \rangle$. GOTO is a two-cycle instruction.

IORLW Inclusive OR literal with W

Syntax: [*label*] IORLW k

Operands: $0 \leq k \leq 255$

Operation: $(W) .\text{OR. } k \rightarrow (W)$

Status Affected: Z

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

INCF Increment f

Syntax: [*label*] INCF f,d

Operands: $0 \leq f \leq 127$
 $d \in [0,1]$

Operation: $(f) + 1 \rightarrow (\text{destination})$

Status Affected: Z

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'.

IORWF Inclusive OR W with f

Syntax: [*label*] IORWF f,d

Operands: $0 \leq f \leq 127$
 $d \in [0,1]$

Operation: $(W) .\text{OR. } (f) \rightarrow (\text{destination})$

Status Affected: Z

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'.

PIC12F752/HV752

| MOVF | Move f |
|------------------|---|
| Syntax: | [<i>label</i>] MOVF f,d |
| Operands: | $0 \leq f \leq 127$ $d \in [0,1]$ |
| Operation: | (f) → (dest) |
| Status Affected: | Z |
| Description: | The contents of register 'f' is moved to a destination dependent upon the status of 'd'. If d = 0, destination is W register. If d = 1, the destination is file register 'f' itself. d = 1 is useful to test a file register since Status flag Z is affected. |
| Words: | 1 |
| Cycles: | 1 |
| <u>Example:</u> | <pre>MOVF FSR, 0</pre> <p>After Instruction</p> <p>W = value in FSR register</p> <p>Z = 1</p> |

| MOVLW | Move literal to W |
|------------------|---|
| Syntax: | [<i>label</i>] MOVLW k |
| Operands: | $0 \leq k \leq 255$ |
| Operation: | $k \rightarrow (W)$ |
| Status Affected: | None |
| Description: | The eight-bit literal 'k' is loaded into W register. The "don't cares" will assemble as '0's. |
| Words: | 1 |
| Cycles: | 1 |
| <u>Example:</u> | <pre>MOVLW 0x5A</pre> <p>After Instruction</p> <p>W = 0x5A</p> |

| MOVWF | Move W to f |
|------------------|---|
| Syntax: | [<i>label</i>] MOVWF f |
| Operands: | $0 \leq f \leq 127$ |
| Operation: | (W) → (f) |
| Status Affected: | None |
| Description: | Move data from W register to register 'f'. |
| Words: | 1 |
| Cycles: | 1 |
| <u>Example:</u> | <pre>MOVWF OPTION F</pre> <p>Before Instruction</p> <p>OPTION = 0xFF W = 0x4F</p> <p>After Instruction</p> <p>OPTION = 0x4F W = 0x4F</p> |

| NOP | No Operation |
|------------------|----------------------|
| Syntax: | [<i>label</i>] NOP |
| Operands: | None |
| Operation: | No operation |
| Status Affected: | None |
| Description: | No operation. |
| Words: | 1 |
| Cycles: | 1 |
| <u>Example:</u> | <pre>NOP</pre> |

RETFIE **Return from Interrupt**

Syntax: [*label*] RETFIE

Operands: None

Operation: TOS → PC,
 1 → GIE

Status Affected: None

Description: Return from Interrupt. Stack is POPed and Top-of-Stack (TOS) is loaded in the PC. Interrupts are enabled by setting Global Interrupt Enable bit, GIE (INTCON<7>). This is a two-cycle instruction.

Words: 1

Cycles: 2

Example: RETFIE

 After Interrupt

 PC = TOS

 GIE = 1

RETLW **Return with literal in W**

Syntax: [*label*] RETLW k

Operands: 0 ≤ k ≤ 255

Operation: k → (W);
 TOS → PC

Status Affected: None

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

 GOTO DONE

TABLE •
 •
 ADDWF PC ;W = offset
 RETLW k1 ;Begin table
 RETLW k2 ;
 •
 •
 •
 RETLW kn ;End of table

 DONE

 Before Instruction
 W = 0x07

 After Instruction
 W = value of k8

RETURN **Return from Subroutine**

Syntax: [*label*] RETURN

Operands: None

Operation: TOS → PC

Status Affected: None

Description: Return from subroutine. The stack is POPed and the top of the stack (TOS) is loaded into the program counter. This is a two-cycle instruction.

PIC12F752/HV752

RLF Rotate Left f through Carry

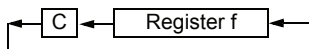
Syntax: `[label] RLF f,d`

Operands: $0 \leq f \leq 127$
 $d \in [0,1]$

Operation: See description below

Status Affected: C

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
    
```

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}

Description: The power-down Status bit, \overline{PD} is cleared. Time-out Status bit, \overline{TO} is set. Watchdog Timer and its prescaler are cleared. The processor is put into Sleep mode with the oscillator stopped.

RRF Rotate Right f through Carry

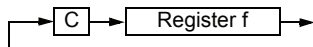
Syntax: `[label] RRF f,d`

Operands: $0 \leq f \leq 127$
 $d \in [0,1]$

Operation: See description below

Status Affected: C

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'.



SUBLW Subtract W from literal

Syntax: `[label] SUBLW k`

Operands: $0 \leq k \leq 255$

Operation: $k - (W) \rightarrow (W)$

Status Affected: C, DC, Z

Description: The W register is subtracted (2's complement method) from the eight-bit literal 'k'. The result is placed in the W register.

| Result | Condition |
|--------|--|
| C = 0 | $W > k$ |
| C = 1 | $W \leq k$ |
| DC = 0 | $W\langle 3:0 \rangle > k\langle 3:0 \rangle$ |
| DC = 1 | $W\langle 3:0 \rangle \leq k\langle 3:0 \rangle$ |

SUBWF **Subtract W from f**

Syntax: [*label*] SUBWF f,d

Operands: $0 \leq f \leq 127$
 $d \in [0,1]$

Operation: $(f) - (W) \rightarrow (\text{destination})$

Status Affected: C, DC, Z

Description: Subtract (2's complement method) 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'.

| | |
|--------|--|
| C = 0 | $W > f$ |
| C = 1 | $W \leq f$ |
| DC = 0 | $W\langle 3:0 \rangle > f\langle 3:0 \rangle$ |
| DC = 1 | $W\langle 3:0 \rangle \leq f\langle 3:0 \rangle$ |

XORWF **Exclusive OR W with f**

Syntax: [*label*] XORWF f,d

Operands: $0 \leq f \leq 127$
 $d \in [0,1]$

Operation: $(W) .XOR. (f) \rightarrow (\text{destination})$

Status Affected: Z

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'.

SWAPF **Swap Nibbles in f**

Syntax: [*label*] SWAPF f,d

Operands: $0 \leq f \leq 127$
 $d \in [0,1]$

Operation: $(f\langle 3:0 \rangle) \rightarrow (\text{destination}\langle 7:4 \rangle)$,
 $(f\langle 7:4 \rangle) \rightarrow (\text{destination}\langle 3:0 \rangle)$

Status Affected: None

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

XORLW **Exclusive OR literal with W**

Syntax: [*label*] XORLW k

Operands: $0 \leq k \leq 255$

Operation: $(W) .XOR. k \rightarrow (W)$

Status Affected: Z

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

PIC12F752/HV752

NOTES:

17.0 SPECIAL FEATURES OF THE CPU

The PIC12F752/HV752 has a host of features intended to maximize system reliability, minimize cost through elimination of external components, provide power-saving features and offer code protection.

These features are:

- Reset
 - Power-on Reset (POR)
 - Power-up Timer (PWRT)
 - Brown-out Reset (BOR)
- Interrupts
- Watchdog Timer (WDT)
- Oscillator selection
- Sleep
- Code protection
- ID Locations
- In-Circuit Serial Programming

The Power-up Timer (PWRT), which provides a fixed delay of 64 ms (nominal) on power-up only, is designed to keep the part in Reset while the power supply stabilizes. There is also circuitry to reset the device if a brown-out occurs, which can use the Power-up Timer to provide at least a 64 ms Reset. With these functions-on-chip, most applications need no external Reset circuitry.

The Sleep mode is designed to offer a very low-current Power-Down mode. The user can wake-up from Sleep through:

- External Reset
- Watchdog Timer Wake-up
- An interrupt

Oscillator selection options are available to allow the part to fit the application. The INTOSC options save system cost, while the External Clock (EC) option provides a means for specific frequency and accurate clock sources. Configuration bits are used to select various options (see [Register 17-1](#)).

17.1 Configuration Bits

The Configuration bits can be programmed (read as '0'), or left unprogrammed (read as '1') to select various device configurations as shown in [Register 17-1](#). These bits are mapped in program memory location 2007h.

Note: Address 2007h is beyond the user program memory space. It belongs to the special configuration memory space (2000h-3FFFh), which can be accessed only during programming. See *Memory Programming Specification* (DS41561) for more information.

PIC12F752/HV752

REGISTER 17-1: CONFIGURATION WORD

| | | | | | |
|--------|----------|----------|-------|------------|-------|
| R/P-1 | R/P-1 | R/P-1 | R/P-1 | R/P-1 | R/P-1 |
| DEBUG | CLKOUTEN | WRT<1:0> | | BOREN<1:0> | |
| bit 13 | | | | bit 8 | |

| | | | | | | | |
|-------|-------|-------|-------|-------|-----|-----|-------|
| U-1 | R/P-1 | R/P-1 | R/P-1 | R/P-1 | U-1 | U-1 | R/P-1 |
| — | CP | MCLRE | PWRTE | WDTE | — | — | FOSC0 |
| bit 7 | | | | | | | bit 0 |

Legend:

R = Readable bit P = Programmable bit U = Unimplemented bit, read as '1'
 '0' = Bit is cleared '1' = Bit is set -n = Value when blank or after Bulk Erase

- bit 13 **DEBUG:** Debug Mode Enable bit⁽²⁾
 1 = Background debugger is disabled
 0 = Background debugger is enabled
- bit 12 **CLKOUTEN:** Clock Out Enable bit
 1 = Clock out function disabled. CLKOUT pin acts as I/O pin
 0 = General purpose I/O disabled. CLKOUT pin acts as CLKOUT
- bit 11-10 **WRT<1:0>:** Flash Program Memory Self Write Enable bit
 11 = Write protection off
 10 = 000h to FFh write-protected, 100h to 3FFh may be modified by PMCON1 control
 01 = 000h to 1FFh write-protected, 200h to 3FFh may be modified by PMCON1 control
 00 = 000h to 3FFh write-protected, entire program is write-protected
- bit 8-9 **BOREN<1:0>:** Brown-out Reset Enable bits
 11 = BOR enabled
 10 = BOR enabled during operation and disabled in Sleep
 0x = BOR disabled
- bit 7 Unimplemented: Read as '1'.
- bit 6 **CP:** Code Protection bit
 1 = Program memory code protection is disabled
 0 = Program memory code protection is enabled
- bit 5 **MCLRE:** MCLR/VPP Pin Function Select bit
 1 = MCLR pin is MCLR function and weak internal pull-up is enabled
 0 = MCLR pin is input function, MCLR function is internally disabled
- bit 4 **PWRTE:** Power-up Timer Enable bit⁽¹⁾
 1 = PWRT disabled
 0 = PWRT enabled
- bit 3 **WDTE:** Watchdog Timer Enable bit
 1 = WDT enabled
 0 = WDT disabled
- bit 2-1 Unimplemented: Read as '1'.
- bit 0 **FOSC:** Oscillator Selection bits
 1 = EC oscillator selected: CLKIN on RA5/CLKIN
 0 = Internal oscillator: I/O function on RA5/CLKIN

Note 1: Enabling Brown-out Reset does not automatically enable Power-up Timer.

Note 2: The Configuration bit is managed automatically by the device development tools. The user should not attempt to manually write this bit location. However, the user should ensure that this location has been programmed to a '1' and the device checksum is correct for proper operation of production software.

17.2 Calibration Bits

The 8 MHz internal oscillator is factory calibrated. These calibration values are stored in fuses located in the Calibration Word (2008h). The Calibration Word is not erased when using the specified bulk erase sequence in the *Memory Programming Specification* (DS41561) and thus, does not require reprogramming.

17.3 Reset

The PIC12F752/HV752 device differentiates between various kinds of Reset:

- Power-on Reset (POR)
- WDT Reset during normal operation
- WDT Reset during Sleep
- MCLR Reset during normal operation
- MCLR Reset during Sleep
- Brown-out Reset (BOR)

Some registers are not affected in any Reset condition; their status is unknown on POR and unchanged in any other Reset. Most other registers are reset to a “Reset state” on:

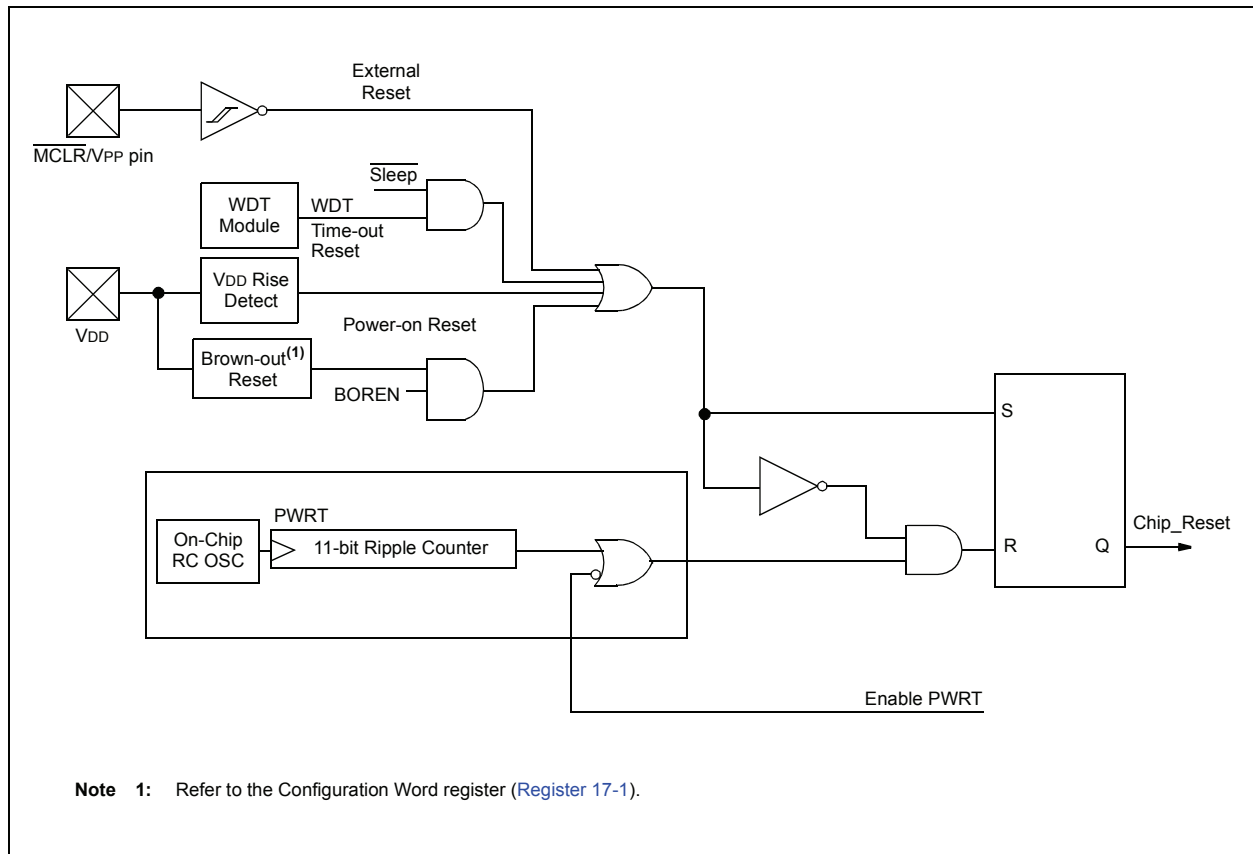
- Power-on Reset
- MCLR Reset
- MCLR Reset during Sleep
- WDT Reset
- Brown-out Reset (BOR)

WDT wake-up does not cause register resets in the same manner as a WDT Reset since wake-up is viewed as the resumption of normal operation. \overline{TO} and \overline{PD} bits are set or cleared differently in different Reset situations, as indicated in Table 17-2. Software can use these bits to determine the nature of the Reset. See Table 17-4 for a full description of Reset states of all registers.

A simplified block diagram of the On-Chip Reset Circuit is shown in Figure 17-1.

The MCLR Reset path has a noise filter to detect and ignore small pulses. See Section 20.0 “Electrical Specifications” for pulse-width specifications.

FIGURE 17-1: SIMPLIFIED BLOCK DIAGRAM OF ON-CHIP RESET CIRCUIT



PIC12F752/HV752

TABLE 17-1: TIME-OUT IN VARIOUS SITUATIONS

| Oscillator Configuration | Power-up | | Brown-out Reset | | Wake-up from Sleep |
|--------------------------|-------------------------------|-------------------------------|-------------------------------|-------------------------------|--------------------|
| | $\overline{\text{PWRTE}} = 0$ | $\overline{\text{PWRTE}} = 1$ | $\overline{\text{PWRTE}} = 0$ | $\overline{\text{PWRTE}} = 1$ | |
| EC, INTOSC | TPWRT | — | TPWRT | — | — |

TABLE 17-2: STATUS/PCON BITS AND THEIR SIGNIFICANCE

| $\overline{\text{POR}}$ | $\overline{\text{BOR}}$ | $\overline{\text{TO}}$ | $\overline{\text{PD}}$ | Condition |
|-------------------------|-------------------------|------------------------|------------------------|--|
| 0 | x | 1 | 1 | Power-on Reset |
| u | 0 | 1 | 1 | Brown-out Reset |
| u | u | 0 | u | WDT Reset |
| u | u | 0 | 0 | WDT Wake-up |
| u | u | u | u | $\overline{\text{MCLR}}$ Reset during normal operation |
| u | u | 1 | 0 | $\overline{\text{MCLR}}$ Reset during Sleep |

Legend: u = unchanged, x = unknown

17.3.1 POWER-ON RESET (POR)

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 POR, simply connect the MCLR pin through a resistor to VDD. This will eliminate external RC components usually needed to create Power-on Reset. A maximum rise time for VDD is required. See [Section 20.0 “Electrical Specifications”](#) for details. If the BOR is enabled, the maximum rise time specification does not apply. The BOR circuitry will keep the device in Reset until VDD reaches VBOR (see [Section 17.3.4 “Brown-out Reset \(BOR\)”](#)).

Note: The POR circuit does not produce an internal Reset when VDD declines. To re-enable the POR, VDD must reach Vss for a minimum of 100 μ s.

When the device starts normal operation (exits the Reset condition), device operating parameters (i.e., voltage, frequency, temperature, etc.) must be met to ensure proper 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 Note AN607, “Power-up Trouble Shooting” (DS00607).

17.3.2 MCLR

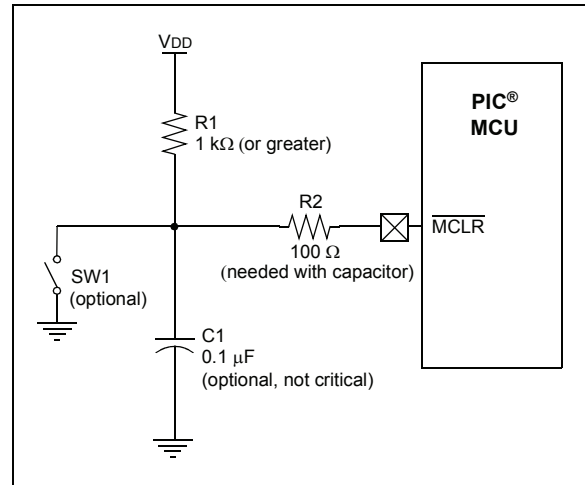
PIC12F752/HV752 has a noise filter in the MCLR Reset path. The filter will detect and ignore small pulses.

It should be noted that a WDT Reset does not drive MCLR pin low.

Voltages applied to the MCLR pin that exceed its specification can result in both MCLR Resets and excessive current beyond the device specification during the ESD event. For this reason, Microchip recommends that the MCLR pin no longer be tied directly to VDD. The use of an RC network, as shown in [Figure 17-2](#), is suggested.

An internal MCLR option is enabled by clearing the MCLRE bit in the Configuration Word register. When MCLRE = 0, the Reset signal to the chip is generated internally. When the MCLRE = 1, the MCLR pin becomes an external Reset input. In this mode, the MCLR pin has a weak pull-up to VDD.

FIGURE 17-2: RECOMMENDED MCLR CIRCUIT



17.3.3 POWER-UP TIMER (PWRT)

The Power-up Timer provides a fixed 64 ms (nominal) time-out on power-up only, from POR or Brown-out Reset. The Power-up Timer operates from an internal RC oscillator. For more information, see [Section 4.2.2 “Internal Clock Mode”](#). The chip is kept in Reset as long as PWRT is active. The PWRT delay allows the VDD to rise to an acceptable level. A Configuration bit, PWRT \overline{E} , can disable (if set) or enable (if cleared or programmed) the Power-up Timer. The Power-up Timer should be enabled when Brown-out Reset is enabled, although it is not required.

The Power-up Timer delay will vary from chip-to-chip due to:

- VDD variation
- Temperature variation
- Process variation

See DC parameters for details ([Section 20.0 “Electrical Specifications”](#)).

Note: Voltage spikes below Vss at the MCLR pin, inducing currents greater than 80 mA, may cause latch-up. Thus, a series resistor of 50-100 Ω should be used when applying a “low” level to the MCLR pin, rather than pulling this pin directly to Vss.

PIC12F752/HV752

17.3.4 BROWN-OUT RESET (BOR)

The BOREN<1:0> bits in the Configuration Word register select one of three BOR modes. One mode has been added to allow control of the BOR enable for lower current during Sleep. By selecting BOREN<1:0> = 10, the BOR is automatically disabled in Sleep to conserve power and enabled on wake-up. See Register 17-1 for the Configuration Word definition.

A brown-out occurs when VDD falls below VBOR for greater than parameter TBOR (see Section 20.0 “Electrical Specifications”). The brown-out condition will reset the device. This will occur regardless of VDD slew rate. A Brown-out Reset may not occur if VDD falls below VBOR for less than parameter TBOR.

On any Reset (Power-on, Brown-out Reset, Watchdog timer, etc.), the chip will remain in Reset until VDD rises above VBOR (see Figure 17-3). If enabled, the Power-up Timer will be invoked by the Reset and keep the chip in Reset an additional 64 ms.

Note: The Power-up Timer is enabled by the $\overline{\text{PWRT}}\overline{\text{E}}$ bit in the Configuration Word register.

If VDD drops below VBOR while the Power-up Timer is running, the chip will go back into a Brown-out Reset and the Power-up Timer will be re-initialized. Once VDD rises above VBOR, the Power-up Timer will execute a 64 ms Reset.

Table 17-3 summarizes the registers associated with BOR.

FIGURE 17-3: BROWN-OUT SITUATIONS

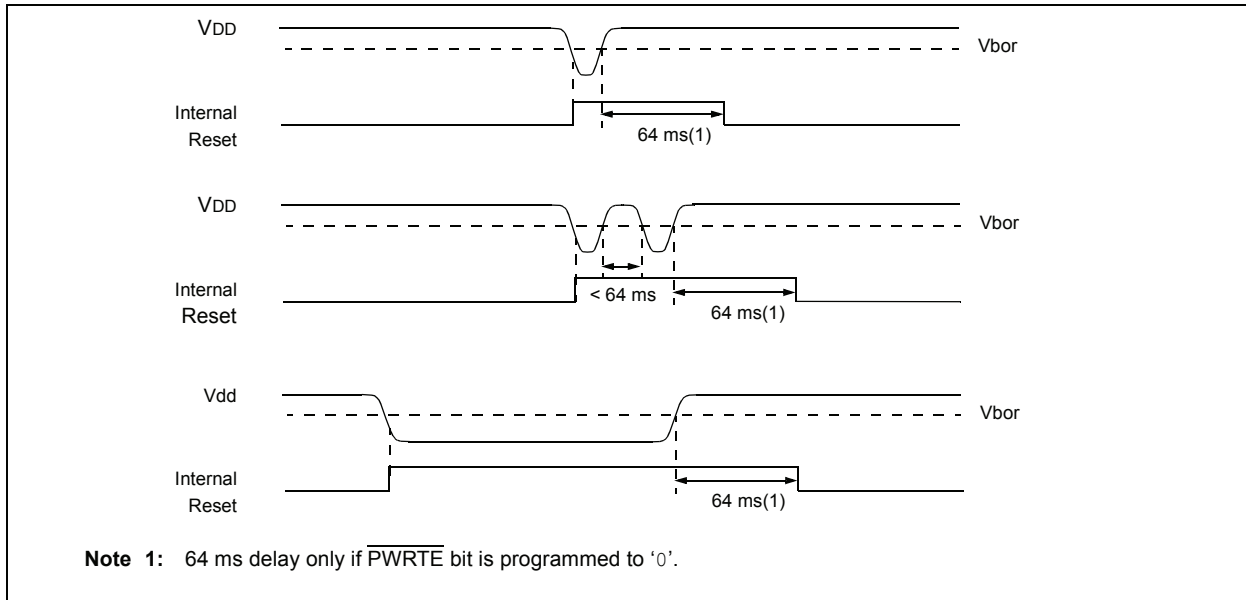


TABLE 17-3: SUMMARY OF REGISTERS ASSOCIATED WITH BROWN-OUT RESET

| Name | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | Register on Page |
|--------|-------|-------|-------|------------------------|------------------------|-------|-------------------------|-------------------------|------------------|
| PCON | — | — | — | — | — | — | $\overline{\text{POR}}$ | $\overline{\text{BOR}}$ | 25 |
| STATUS | IRP | RP1 | RP0 | $\overline{\text{TO}}$ | $\overline{\text{PD}}$ | Z | DC | C | 18 |

Legend: u = unchanged, x = unknown, — = unimplemented bit, reads as '0', q = value depends on condition. Shaded cells are not used by BOR.

Note 1: Other (non Power-up) Resets include $\overline{\text{MCLR}}$ Reset and Watchdog Timer Reset during normal operation.

17.3.5 TIME-OUT SEQUENCE

On power-up, the time-out sequence is as follows:

- PWRT time-out is invoked after POR has expired.
- OST is activated after the PWRT time-out has expired.

The total time-out will vary based on oscillator configuration and $\overline{\text{PWRT}}\text{E}$ bit status. For example, in EC mode with $\overline{\text{PWRT}}\text{E}$ bit erased ($\overline{\text{PWRT}}\text{E}$ disabled), there will be no time-out at all. Figure 17-4, Figure 17-5 and Figure 17-6 depict time-out sequences.

Since the time-outs occur from the POR pulse, if $\overline{\text{MCLR}}$ is kept low long enough, the time-outs will expire. Then, bringing $\overline{\text{MCLR}}$ high will begin execution immediately (see Figure 17-5). This is useful for testing purposes or to synchronize more than one PIC12F752/HV752 device operating in parallel.

Table 17-5 shows the Reset conditions for some special registers, while Table 17-4 shows the Reset conditions for all the registers.

17.3.6 POWER CONTROL (PCON) REGISTER

The Power Control register PCON (address 8Eh) has two Status bits to indicate what type of Reset occurred last.

Bit 0 is $\overline{\text{BOR}}$ (Brown-out). $\overline{\text{BOR}}$ is unknown on Power-on Reset. It must then be set by the user and checked on subsequent Resets to see if $\overline{\text{BOR}} = 0$, indicating that a Brown-out has occurred. The $\overline{\text{BOR}}$ Status bit is a “don’t care” and is not necessarily predictable if the brown-out circuit is disabled ($\text{BOREN} < 1:0 > = 00$ in the Configuration Word register).

Bit 1 is $\overline{\text{POR}}$ (Power-on Reset). It is a ‘0’ on Power-on Reset and unaffected otherwise. The user must write a ‘1’ to this bit following a Power-on Reset. On a subsequent Reset, if $\overline{\text{POR}}$ is ‘0’, it will indicate that a Power-on Reset has occurred (i.e., VDD may have gone too low).

For more information, see Section 17.3.4 “Brown-out Reset (BOR)”.

FIGURE 17-4: TIME-OUT SEQUENCE ON POWER-UP (DELAYED $\overline{\text{MCLR}}$): CASE 1

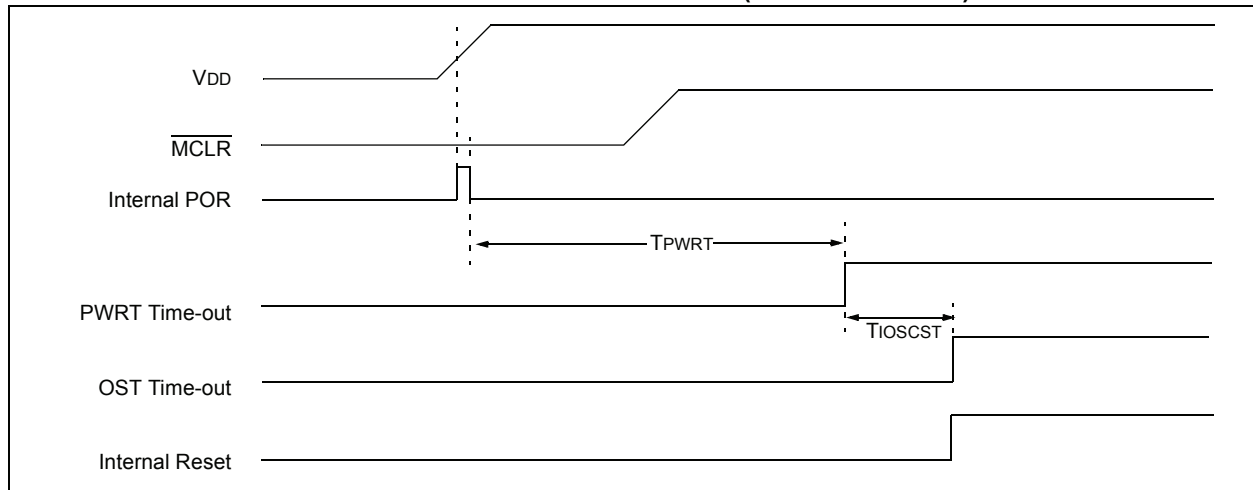
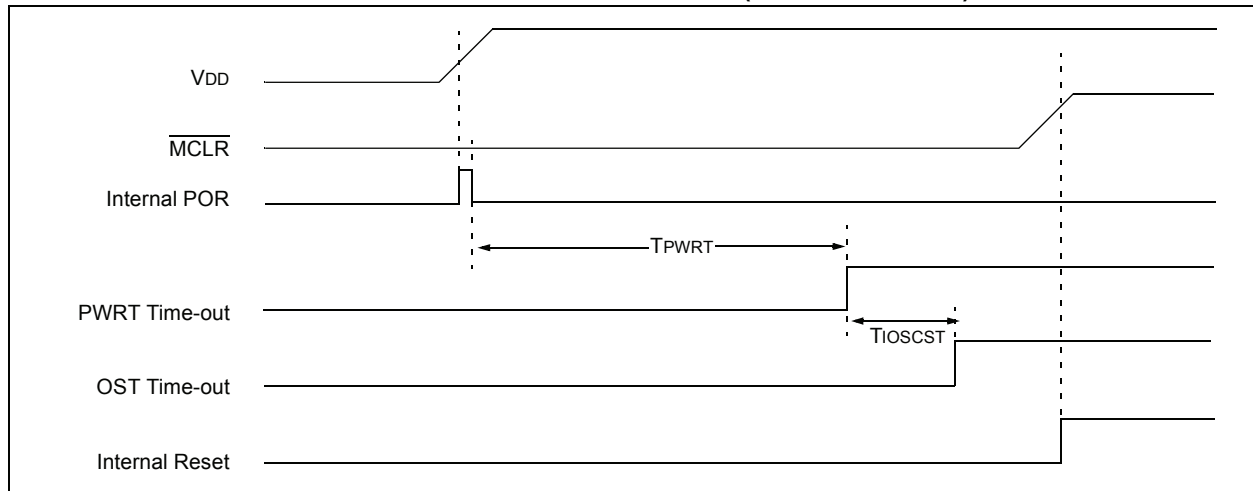


FIGURE 17-5: TIME-OUT SEQUENCE ON POWER-UP (DELAYED $\overline{\text{MCLR}}$): CASE 2



PIC12F752/HV752

FIGURE 17-6: TIME-OUT SEQUENCE ON POWER-UP ($\overline{\text{MCLR}}$ WITH V_{DD})

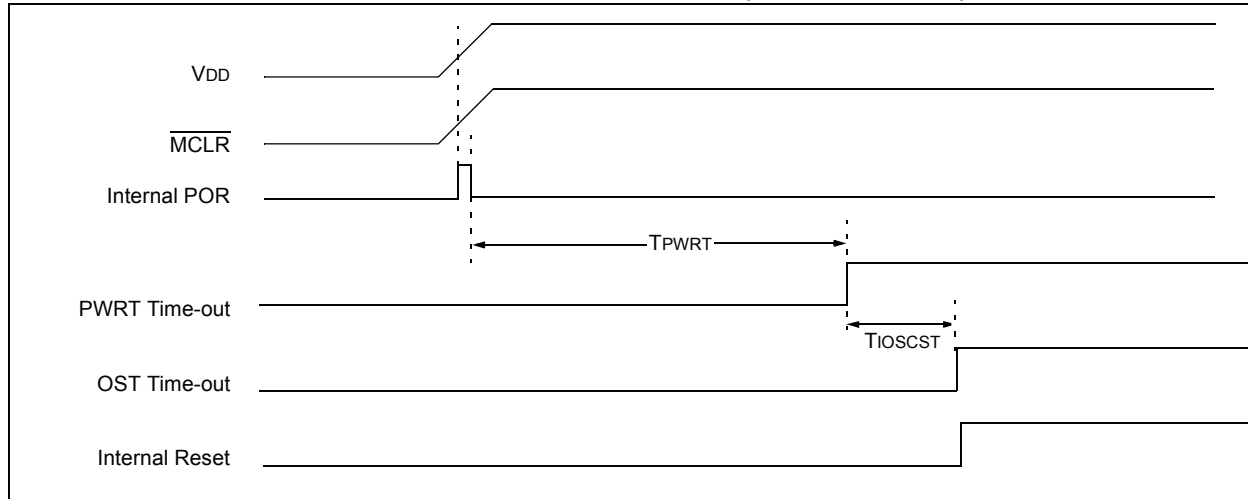


TABLE 17-4: INITIALIZATION CONDITION FOR REGISTERS

| Register | Address | Power-on Reset | MCLR Reset WDT Reset Brown-out Reset ⁽¹⁾ | Wake-up from Sleep through Interrupt Wake-up from Sleep through WDT Time-out |
|------------------------|-----------------------|----------------|---|---|
| W | — | xxxx xxxx | uuuu uuuu | uuuu uuuu |
| INDF | 00h/80h/ 100h/180h | xxxx xxxx | xxxx xxxx | uuuu uuuu |
| TMR0 | 01h | xxxx xxxx | uuuu uuuu | uuuu uuuu |
| PCL | 02h/82h/ 102h/182h | 0000 0000 | 0000 0000 | PC + 1 ⁽³⁾ |
| STATUS | 03h/83h/ 103h/183h | 0001 1xxx | 000q quuu ⁽⁴⁾ | uuuq quuu ⁽⁴⁾ |
| FSR | 04h/84h/ 104h/184h | xxxx xxxx | uuuu uuuu | uuuu uuuu |
| PORTA | 05h | --xx xxxx | --uu uuuu | --uu uuuu |
| IOCAF | 08h | --00 0000 | --00 0000 | --uu uuuu |
| PCLATH | 0Ah/8Ah/ 10Ah/18Ah | ---0 0000 | ---0 0000 | ---u uuuu |
| INTCON | 0Bh/8Bh/ 10Bh/18Bh | 0000 0000 | 0000 0000 | uuuu uuuu ⁽²⁾ |
| PIR1 | 0Ch | 00-- -0-0 | 00-- -0-0 | uu-- -u-u ⁽²⁾ |
| PIR2 | 0Dh | --00 -0-0 | --00 -0-0 | --uu -u-u ⁽²⁾ |
| TMR1L | 0Fh | xxxx xxxx | uuuu uuuu | uuuu uuuu |
| TMR1H | 10h | xxxx xxxx | uuuu uuuu | uuuu uuuu |
| T1CON | 11h | 0000 00-0 | uuuu uu-u | uuuu uu-u |
| T1GCON | 12h | 0000 0x00 | 0000 0x00 | uuuu uuuu |
| CCPR1L ⁽¹⁾ | 13h | xxxx xxxx | uuuu uuuu | uuuu uuuu |
| CCPR1H ⁽¹⁾ | 14h | xxxx xxxx | uuuu uuuu | uuuu uuuu |
| CCP1CON ⁽¹⁾ | 15h | --00 0000 | --00 0000 | --uu uuuu |
| ADRESL ⁽¹⁾ | 1Ch | xxxx xxxx | uuuu uuuu | uuuu uuuu |
| ADRESH ⁽¹⁾ | 1Dh | xxxx xxxx | uuuu uuuu | uuuu uuuu |
| ADCON0 ⁽¹⁾ | 1Eh | 0000 0000 | 0000 0000 | uuuu uuuu |
| ADCON1 ⁽¹⁾ | 1Fh | -000 ---- | -000 ---- | -uuu ---- |
| OPTION_REG | 81h/181h | 1111 1111 | 1111 1111 | uuuu uuuu |
| TRISA | 85h | --11 1111 | --11 1111 | --uu uuuu |
| IOCAP | 88h | --00 0000 | --00 0000 | --uu uuuu |
| PIE1 | 8Ch | 00-- -000 | 00-- -000 | uu-- -uuu |
| PIE2 | 8Dh | --00 -0-0 | --00 -0-0 | --uu -u-u |
| OSCCON | 8Fh | --01 -00- | --uu -uu- | --uu -uu- |
| FVRCON | 90h | 0000 ---- | 0000 ---- | uuuu ---- |
| DACCON0 | 91h | 000- -0-- | 000- -0-- | uuu- -u-- |
| DACCON1 | 92h | ---0 0000 | ---0 0000 | ---u uuuu |
| CM2CON0 | 9Bh | 0000 0100 | 0000 0100 | uuuu uuuu |

Legend: u = unchanged, x = unknown, - = unimplemented bit, reads as '0', q = value depends on condition.

Note 1: If VDD goes too low, Power-on Reset will be activated and registers will be affected differently.

2: One or more bits in INTCON and/or PIRx will be affected (to cause wake-up).

3: When the wake-up is due to an interrupt and the GIE bit is set, the PC is loaded with the interrupt vector (0004h).

4: See Table 17-5 for Reset value for specific condition.

5: If Reset was due to brown-out, then bit 0 = 0. All other Resets will cause bit 0 = u.

PIC12F752/HV752

TABLE 17-4: INITIALIZATION CONDITION FOR REGISTERS (CONTINUED)

| Register | Address | Power-on Reset | MCLR Reset WDT Reset (Continued) Brown-out Reset ⁽¹⁾ | Wake-up from Sleep through Interrupt Wake-up from Sleep through WDT Time-out (Continued) |
|----------|---------|----------------|---|---|
| CM2CON1 | 9Ch | 0000 ---0 | 0000 ---0 | uuuu ---u |
| CM1CON0 | 9Dh | 0000 0100 | 0000 0100 | uuuu uuuu |
| CM1CON1 | 9Eh | 0000 ---0 | 0000 ---0 | uuuu ---u |
| CMOUT | 9Fh | ---- --00 | ---- --00 | ---- --uu |
| LATA | 105h | --xx -xxx | --uu -uuu | --uu -uuu |
| IOCAN | 108h | --00 0000 | --00 0000 | --uu uuuu |
| WPUA | 10Ch | --00 0000 | --00 0000 | --uu uuuu |
| SLRCON0 | 10Dh | ---- -0-0 | ---- -0-0 | ---- -u-u |
| PCON | 10Fh | ---- --qq | ---- --uu ^(1, 5) | ---- --uu |
| TMR2 | 110h | 0000 0000 | 0000 0000 | uuuu uuuu |
| PR2 | 111h | 1111 1111 | 1111 1111 | uuuu uuuu |
| T2CON | 112h | -000 0000 | -000 0000 | -uuu uuuu |
| HLTMR1 | 113h | 0000 0000 | 0000 0000 | uuuu uuuu |
| HLTPR1 | 114h | 1111 1111 | 1111 1111 | uuuu uuuu |
| HLT1CON0 | 115h | -000 0000 | -000 0000 | -uuu uuuu |
| HLT1CON1 | 116h | ---0 0000 | ---0 0000 | ---u uuuu |
| ANSELA | 185h | --11 -111 | --11 -111 | --uu -uuu |
| APFCON | 188h | ---0 -000 | ---0 -000 | ---u -uuu |
| OSCTUNE | 189h | ---0 0000 | ---u uuuu | ---u uuuu |
| PMCON1 | 18Ch | ---- -000 | ---- -000 | ---- -uuu |
| PMCON2 | 18Dh | ---- ---- | ---- ---- | ---- ---- |
| PMADRL | 18Eh | 0000 0000 | 0000 0000 | uuuu uuuu |
| PMADRH | 18Fh | ---- --00 | ---- --00 | ---- --uu |
| PMDATL | 190h | 0000 0000 | 0000 0000 | uuuu uuuu |
| PMDATH | 191h | --00 0000 | --00 0000 | --uu uuuu |
| COG1PH | 192h | ---- xxxx | ---- uuuu | ---- uuuu |
| COG1BLK | 193h | xxxx xxxx | uuuu uuuu | uuuu uuuu |
| COG1DB | 194h | xxxx xxxx | uuuu uuuu | uuuu uuuu |
| COG1CON0 | 195h | 0000 0000 | 0000 0000 | uuuu uuuu |
| COG1CON1 | 196h | --00 0000 | --00 0000 | --uu uuuu |
| COG1ASD | 197h | 0000 0000 | 0000 0000 | uuuu uuuu |

Legend: u = unchanged, x = unknown, - = unimplemented bit, reads as '0', q = value depends on condition.

Note 1: If VDD goes too low, Power-on Reset will be activated and registers will be affected differently.

2: One or more bits in INTCON and/or PIRx will be affected (to cause wake-up).

3: When the wake-up is due to an interrupt and the GIE bit is set, the PC is loaded with the interrupt vector (0004h).

4: See Table 17-5 for Reset value for specific condition.

5: If Reset was due to brown-out, then bit 0 = 0. All other Resets will cause bit 0 = u.

TABLE 17-5: INITIALIZATION CONDITION FOR SPECIAL REGISTERS

| Condition | Program Counter | Status Register | PCON Register |
|------------------------------------|-----------------------|-----------------|---------------|
| Power-on Reset | 000h | 0001 1xxx | ---- --0x |
| MCLR Reset during normal operation | 000h | 000u uuuu | ---- --uu |
| MCLR Reset during Sleep | 000h | 0001 0uuu | ---- --uu |
| WDT Reset | 000h | 0000 uuuu | ---- --uu |
| WDT Wake-up | PC + 1 | uuu0 0uuu | ---- --uu |
| Brown-out Reset | 000h | 0001 1uuu | ---- --u0 |
| Interrupt Wake-up from Sleep | PC + 1 ⁽¹⁾ | uuu1 0uuu | ---- --uu |

Legend: u = unchanged, x = unknown, – = unimplemented bit, reads as ‘0’.

Note 1: When the wake-up is due to an interrupt and Global Interrupt Enable bit, GIE, is set, the PC is loaded with the interrupt vector (0004h) after execution of PC + 1.

PIC12F752/HV752

17.4 Interrupts

The PIC12F752/HV752 has multiple sources of interrupt:

- External Interrupt (INT pin)
- Interrupt-On-Change (IOC) Interrupts
- Timer0 Overflow Interrupt
- Timer1 Overflow Interrupt
- Timer2 Match Interrupt
- Hardware Limit Timer (HLT) Interrupt
- Comparator Interrupt (C1/C2)
- ADC Interrupt
- Complementary Output Generator (COG)
- CCP1 Interrupt
- Flash Memory Self Write

The Interrupt Control register (INTCON) and Peripheral Interrupt Request Registers (PIRx) record individual interrupt requests in flag bits. The INTCON register also has individual and global interrupt enable bits.

The Global Interrupt Enable bit, GIE of the INTCON register, enables (if set) all unmasked interrupts, or disables (if cleared) all interrupts. Individual interrupts can be disabled through their corresponding enable bits in the INTCON register and PIRx registers. GIE is cleared on Reset.

When an interrupt is serviced, the following actions occur automatically:

- The GIE is cleared to disable any further interrupt.
- The return address is pushed onto the stack.
- The PC is loaded with 0004h.

The Return from Interrupt instruction, `RETFIE`, exits the interrupt routine, as well as sets the GIE bit, which re-enables unmasked interrupts.

The following interrupt flags are contained in the INTCON register:

- INT Pin Interrupt
- Interrupt-On-Change (IOC) Interrupts
- Timer0 Overflow Interrupt

The peripheral interrupt flags are contained in the PIR1 and PIR2 registers. The corresponding interrupt enable bit is contained in the PIE1 and PIE2 registers.

The following interrupt flags are contained in the PIR1 register:

- A/D Interrupt
- Comparator Interrupt
- Timer1 Overflow Interrupt
- Timer2 Match Interrupt
- Enhanced CCP Interrupt

For external interrupt events, such as the INT pin or PORTA change interrupt, the interrupt latency will be three or four instruction cycles. The exact latency depends upon when the interrupt event occurs (see

Figure 17-8). The latency is the same for one or two-cycle instructions. Once in the Interrupt Service Routine, the source(s) of the interrupt can be determined by polling the interrupt flag bits. The interrupt flag bit(s) must be cleared in software before re-enabling interrupts to avoid multiple interrupt requests.

Note 1: Individual interrupt flag bits are set, regardless of the status of their corresponding mask bit or the GIE bit.

2: When an instruction that clears the GIE bit is executed, any interrupts that were pending for execution in the next cycle are ignored. The interrupts, which were ignored, are still pending to be serviced when the GIE bit is set again.

For additional information on Timer1, Timer2, comparators, ADC, Enhanced CCP modules, refer to the respective peripheral section.

17.4.1 RA2/INT INTERRUPT

The external interrupt on the RA2/INT pin is edge-triggered; either on the rising edge if the INTEDG bit of the OPTION register is set, or the falling edge, if the INTEDG bit is clear. When a valid edge appears on the RA2/INT pin, the INTF bit of the INTCON register is set. This interrupt can be disabled by clearing the INTE control bit of the INTCON register. The INTF bit must be cleared by software in the Interrupt Service Routine before re-enabling this interrupt. The RA2/INT interrupt can wake-up the processor from Sleep, if the INTE bit was set prior to going into Sleep. See [Section 17.7 “Power-Down Mode \(Sleep\)”](#) for details on Sleep and [Figure 17-10](#) for timing of wake-up from Sleep through RA2/INT interrupt.

Note: The ANSEL register must be initialized to configure an analog channel as a digital input. Pins configured as analog inputs will read '0' and cannot generate an interrupt.

17.4.2 TIMER0 INTERRUPT

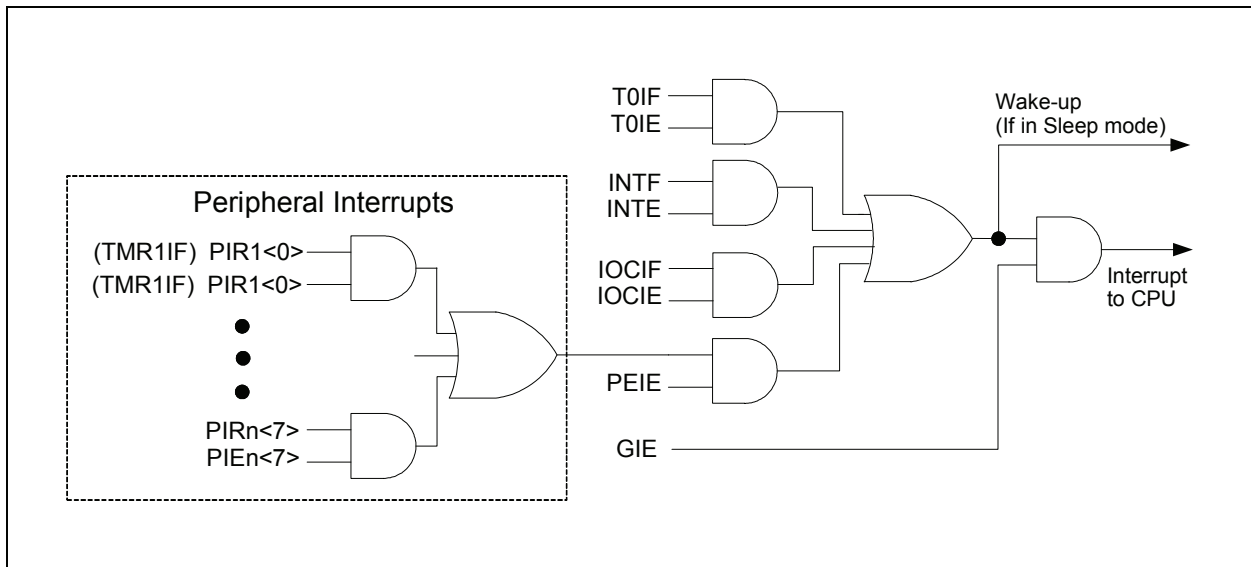
An overflow (FFh → 00h) in the TMR0 register will set the TOIF bit of the INTCON register. The interrupt can be enabled/disabled by setting/clearing TOIE bit of the INTCON register. See [Section 6.0 “Timer0 Module”](#) for operation of the Timer0 module.

17.4.3 PORTA INTERRUPT-ON-CHANGE

An input change on PORTA sets the IOCIF bit of the INTCON register. The interrupt can be enabled/disabled by setting/clearing the IOCIE bit of the INTCON register. Plus, individual pins can be configured through the IOC register.

Note: If a change on the I/O pin should occur when any PORTA operation is being executed, then the IOCIF interrupt flag may not get set.

FIGURE 17-7: INTERRUPT LOGIC



PIC12F752/HV752

FIGURE 17-8: INT PIN INTERRUPT TIMING

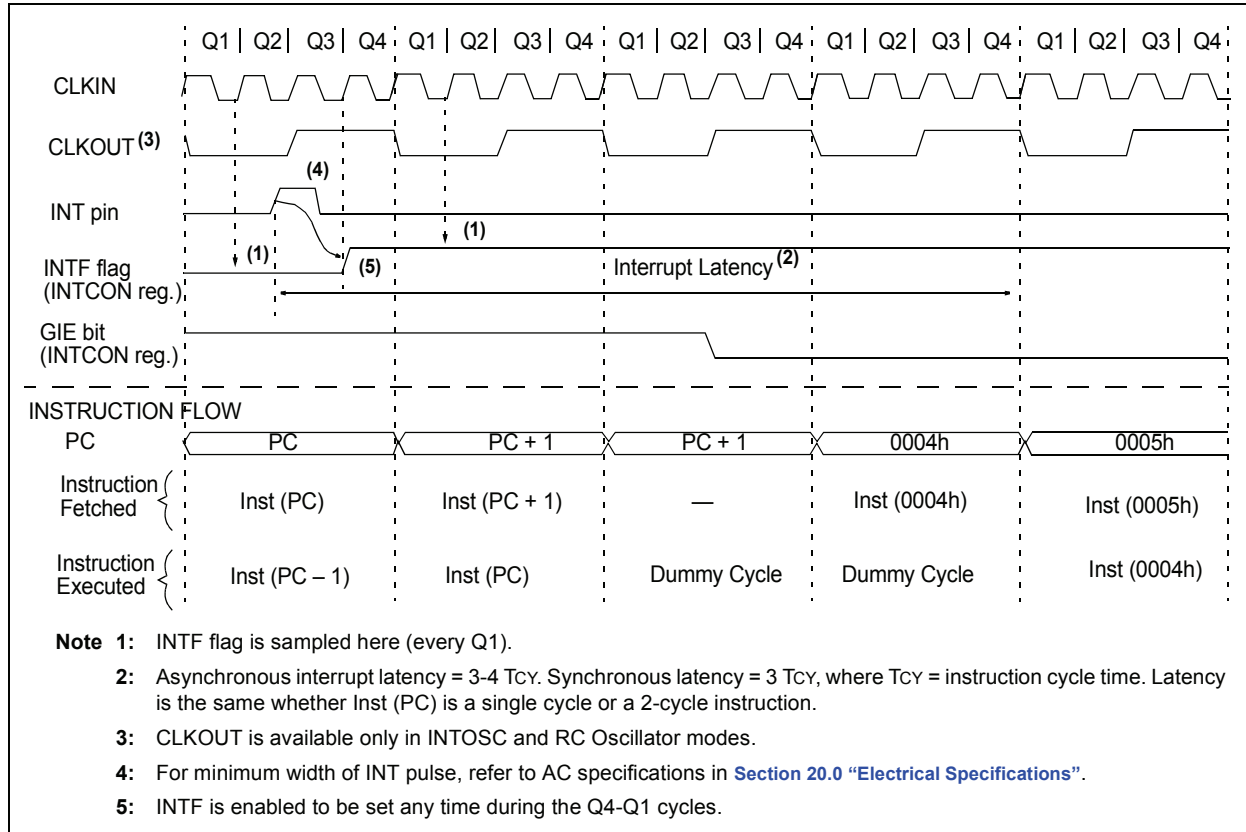


TABLE 17-6: SUMMARY OF REGISTERS ASSOCIATED WITH INTERRUPTS

| Name | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | Register on Page |
|--------|---------|-------|--------|--------|--------|----------|--------|--------|------------------|
| INTCON | GIE | PEIE | T0IE | INTE | IOCIE | T0IF | INTF | IOCIF | 20 |
| IOCAF | — | — | IOCAF5 | IOCAF4 | IOCAF3 | IOCAF2 | IOCAF1 | IOCAF0 | 52 |
| IOCAN | — | — | IOCAN5 | IOCAN4 | IOCAN3 | IOCAN2 | IOCAN1 | IOCAN0 | 52 |
| IOCAP | — | — | IOCAP5 | IOCAP4 | IOCAP3 | IOCAP2 | IOCAP1 | IOCAP0 | 52 |
| LATA | — | — | LATA5 | LATA4 | — | LATA2 | LATA1 | LATA0 | 48 |
| PIE1 | TMR1GIE | ADIE | — | — | — | HLTMR1IE | TMR2IE | TMR1IE | 21 |
| PIR1 | TMR1GIF | ADIF | — | — | — | HLTMR1IF | TMR2IF | TMR1IF | 23 |

Legend: x = unknown, u = unchanged, — = unimplemented read as '0', q = value depends upon condition. Shaded cells are not used by the interrupt module.

17.5 Context Saving During Interrupts

During an interrupt, only the return PC value is saved on the stack. Typically, users may wish to save key registers during an interrupt (e.g., W and STATUS registers). This must be implemented in software.

Temporary holding registers W_TEMP and STATUS_TEMP should be placed in the last 16 bytes of GPR (see Figure 2-2). These 16 locations are common to all banks and do not require banking. This makes context save and restore operations simpler. The code shown in Example 17-1 can be used to:

- Store the W register
- Store the STATUS register
- Execute the ISR code
- Restore the Status (and Bank Select Bit register)
- Restore the W register

Note: The PIC12F752/HV752 does not require saving the PCLATH. However, if computed GOTOS are used in both the ISR and the main code, the PCLATH must be saved and restored in the ISR.

EXAMPLE 17-1: SAVING STATUS AND W REGISTERS IN RAM

```

MOVWF  W_TEMP           ;Copy W to TEMP register
SWAPF  STATUS,W        ;Swap status to be saved into W
                        ;Swaps are used because they do not affect the status bits
MOVWF  STATUS_TEMP     ;Save status to bank zero STATUS_TEMP register
:
: (ISR)                 ;Insert user code here
:
SWAPF  STATUS_TEMP,W   ;Swap STATUS_TEMP register into W
                        ;(sets bank to original state)
MOVWF  STATUS          ;Move W into STATUS register
SWAPF  W_TEMP,F        ;Swap W_TEMP
SWAPF  W_TEMP,W        ;Swap W_TEMP into W
    
```

17.6 Watchdog Timer (WDT)

The Watchdog Timer is a free running timer, using LFINTOSC oscillator as its clock source. The WDT is enabled by setting the WDTE bit of the Configuration Word (default setting). When WDTE is set, the LFINTOSC will always be enabled to provide a clock source to the WDT module.

During normal operation, a WDT time-out generates a device Reset. If the device is in Sleep mode, a WDT time-out causes the device to wake-up and continue with normal operation.

The WDT can be permanently disabled by programming the Configuration bit, WDTE, as clear (Section 17.1 “Configuration Bits”).

17.6.1 WDT PERIOD

The WDT has a nominal time-out period of 18 ms (with no prescaler). The time-out periods vary with temperature, VDD and process variations from part to part (see DC specs). If longer time-out periods are 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, time-out periods up to 2.3 seconds can be realized.

The CLRWDT and SLEEP instructions clear the WDT and the prescaler, if assigned to the WDT, and prevent it from timing out and generating a device Reset.

The \overline{TO} bit in the STATUS register will be cleared upon a Watchdog Timer time-out.

PIC12F752/HV752

17.6.2 WDT PROGRAMMING CONSIDERATIONS

It should also be taken in account that under worst-case conditions (i.e., $V_{DD} = \text{Min.}$, Temperature = Max., Max. WDT prescaler) it may take several seconds before a WDT time-out occurs.

FIGURE 17-9: WATCHDOG TIMER WITH SHARED PRESCALE BLOCK DIAGRAM

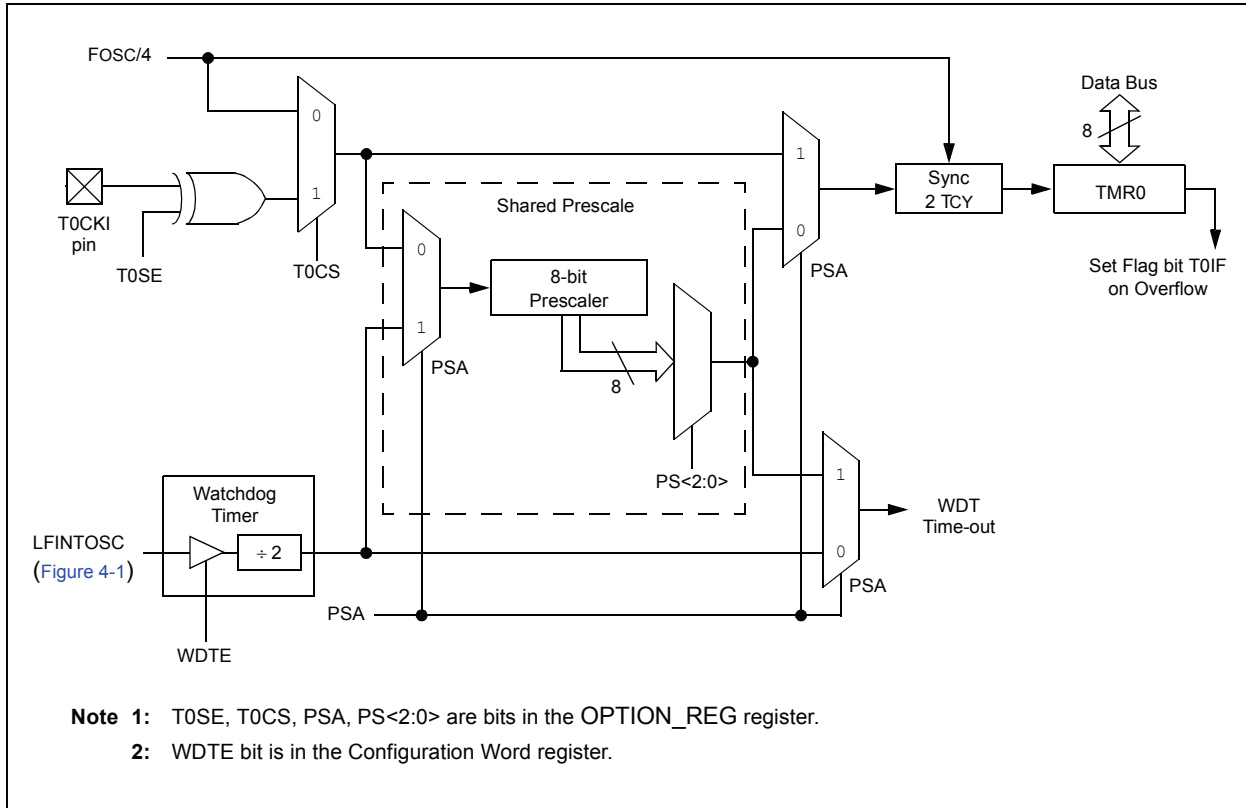


TABLE 17-7: WDT STATUS

| Conditions | WDT |
|----------------|---------|
| WDTE = 0 | Cleared |
| CLRWDT Command | |
| Exit Sleep | |

TABLE 17-8: SUMMARY OF REGISTERS ASSOCIATED WITH WATCHDOG TIMER

| Name | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | Register on Page |
|------------|--------------------------|--------|-------|-------|-------|---------|-------|-------|------------------|
| OPTION_REG | $\overline{\text{RAPU}}$ | INTEDG | T0CS | T0SE | PSA | PS<2:0> | | | 57 |

Legend: Shaded cells are not used by the Watchdog Timer.

Note 1: See [Register 17-1](#) for operation of all Configuration Word register bits.

TABLE 17-9: SUMMARY OF CONFIGURATION WORD WITH WATCHDOG TIMER

| Name | Bits | Bit -/7 | Bit -/6 | Bit 13/5 | Bit 12/4 | Bit 11/3 | Bit 10/2 | Bit 9/1 | Bit 8/0 | Register on Page |
|--------|------|---------|------------------------|---------------------------|------------------------------|----------|----------|------------|---------|------------------|
| CONFIG | 13:8 | — | — | $\overline{\text{DEBUG}}$ | $\overline{\text{CLKOUTEN}}$ | WRT<1:0> | | BOREN<1:0> | | 142 |
| | 7:0 | — | $\overline{\text{CP}}$ | MCLRE | $\overline{\text{PWRTE}}$ | WDTE | — | — | FOSC0 | |

Legend: — = unimplemented location, read as '1'. Shaded cells are not used by Watchdog Timer.

PIC12F752/HV752

17.7 Power-Down Mode (Sleep)

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

If the Watchdog Timer is enabled:

- $\overline{\text{WDT}}$ will be cleared but keeps running.
- $\overline{\text{PD}}$ bit in the STATUS register is cleared.
- $\overline{\text{TO}}$ bit is set.
- Oscillator driver is turned off.
- I/O ports maintain the status they had before `SLEEP` was executed (driving high, low or high-impedance).

For lowest current consumption in this mode, all I/O pins should be either at V_{DD} or V_{SS} , with no external circuitry drawing current from the I/O pin and the comparators, DAC and FVR should be disabled. I/O pins that are high-impedance inputs should be pulled high or low externally to avoid switching currents caused by floating inputs. The $\overline{\text{T0CKI}}$ input should also be at V_{DD} or V_{SS} for lowest current consumption. The contribution from on-chip pull-ups on PORTA should be considered.

The $\overline{\text{MCLR}}$ pin must be at a logic high level.

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

17.7.1 WAKE-UP FROM SLEEP

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

1. External Reset input on $\overline{\text{MCLR}}$ pin.
2. Watchdog Timer wake-up.
3. Interrupt from INT pin.
4. Interrupt-On-Change input change.
5. Peripheral interrupt.

The first event will cause a device Reset. The other events are considered a continuation of program execution. The $\overline{\text{TO}}$ and $\overline{\text{PD}}$ bits in the STATUS register can be used to determine the cause of device Reset. The $\overline{\text{PD}}$ bit, which is set on power-up, is cleared when Sleep is invoked. $\overline{\text{TO}}$ bit is cleared if WDT wake-up occurred.

The following peripheral interrupts can wake the device from Sleep:

1. Timer1 interrupt. Timer1 must be operating as an asynchronous counter.
2. CCP Capture mode interrupt.
3. A/D conversion (when A/D clock source is RC).
4. Comparator output changes state.
5. Interrupt-on-change.
6. External Interrupt from INT pin.

Other peripherals cannot generate interrupts since during Sleep, no on-chip clocks are present.

When the `SLEEP` instruction is being executed, the next instruction ($\text{PC} + 1$) is prefetched. For the device to wake-up through an interrupt event, the corresponding interrupt enable bit must be set (enabled). Wake-up is regardless of the state of the GIE bit. If the GIE bit is clear (disabled), the device continues execution at the instruction after the `SLEEP` instruction. If the GIE bit is set (enabled), the device executes the instruction after the `SLEEP` instruction, then branches to the interrupt address (0004h). In cases where the execution of the instruction following `SLEEP` is not desirable, the user should have a `NOP` after the `SLEEP` instruction.

Note: If the global interrupts are disabled (GIE is cleared) and any interrupt source has both its interrupt enable bit and the corresponding interrupt flag bits set, the device will immediately wake-up from Sleep.

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

17.7.2 WAKE-UP USING INTERRUPTS

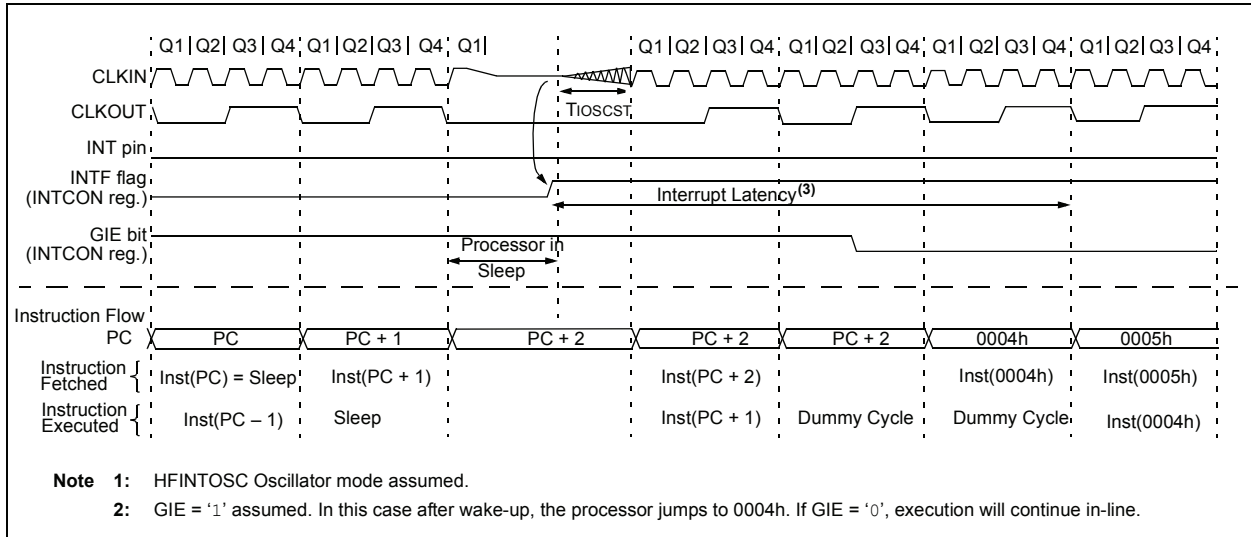
When global interrupts are disabled (GIE cleared) and any interrupt source has both its interrupt enable bit and interrupt flag bit set, one of the following will occur:

- If the interrupt occurs **before** the execution of a `SLEEP` instruction, the `SLEEP` instruction will complete as a `NOP`. Therefore, the WDT and WDT prescaler and postscaler (if enabled) will not be cleared, the $\overline{\text{TO}}$ bit will not be set and the $\overline{\text{PD}}$ bit will not be cleared.
- If the interrupt occurs **during or after** the execution of a `SLEEP` instruction, the device will immediately wake-up from Sleep. The `SLEEP` instruction is executed. Therefore, the WDT and WDT prescaler and postscaler (if enabled) will be cleared, the $\overline{\text{TO}}$ bit will be set and the $\overline{\text{PD}}$ bit will be cleared.

Even if the flag bits were checked before executing a `SLEEP` instruction, it may be possible for flag bits to become set before the `SLEEP` instruction completes. To determine whether a `SLEEP` instruction executed, test the $\overline{\text{PD}}$ bit. If the $\overline{\text{PD}}$ bit is set, the `SLEEP` instruction was executed as a `NOP`.

To ensure that the WDT is cleared, a `CLRWDT` instruction should be executed before a `SLEEP` instruction. See [Figure 17-10](#) for more details.

FIGURE 17-10: WAKE-UP FROM SLEEP THROUGH INTERRUPT



17.8 Code Protection

If the code protection bit(s) have not been programmed, the on-chip program memory can be read out using ICSP™ for verification purposes.

Note: The entire Flash program memory will be erased when the code protection is turned off. See the “Program Memory Specification” (DS41561) for more information.

17.9 ID Locations

Four memory locations (2000h-2003h) 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 mode. Only the Least Significant 7 bits of the ID locations are reported when using MPLAB® IDE.

PIC12F752/HV752

17.10 In-Circuit Serial Programming™

The PIC12F752/HV752 microcontrollers can be serially programmed while in the end application circuit. This is simply done with five connections for:

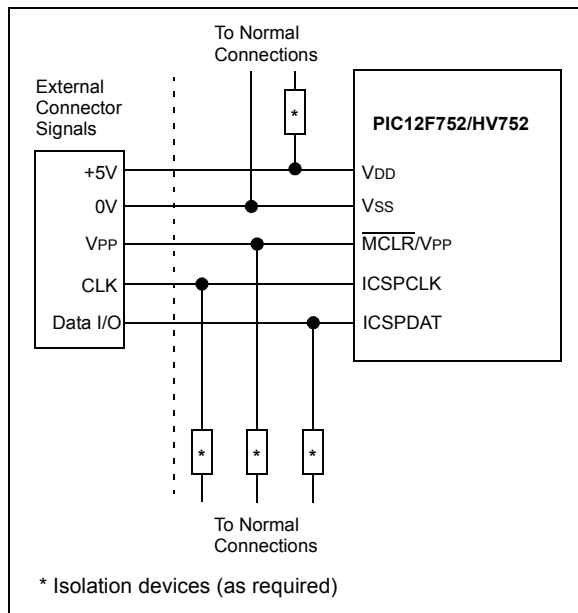
- clock
- data
- power
- ground
- programming voltage

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 $\overline{\text{ICSPDAT}}$ and ICSPCLK pins low, while raising the $\overline{\text{MCLR}}$ (V_{PP}) pin from V_{IL} to V_{IH} . See the *Memory Programming Specification* (DS41561) for more information. ICSPDAT becomes the programming data and ICSPCLK becomes the programming clock. Both ICSPDAT and ICSPCLK are Schmitt Trigger inputs in Program/Verify mode.

A typical In-Circuit Serial Programming connection is shown in Figure 17-11.

FIGURE 17-11: TYPICAL IN-CIRCUIT SERIAL PROGRAMMING CONNECTION



Note: To erase the device, V_{DD} must be above the Bulk Erase V_{DD} minimum given in the “Program Memory Specification” (DS41561)

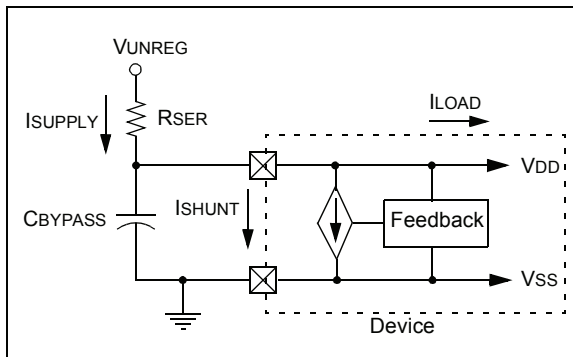
18.0 SHUNT REGULATOR (PIC12HV752 ONLY)

The PIC12HV752 devices include a permanent internal 5 volt (nominal) shunt regulator in parallel with the VDD pin. This eliminates the need for an external voltage regulator in systems sourced by an unregulated supply. All external devices connected directly to the VDD pin will share the regulated supply voltage and contribute to the total VDD supply current (ILOAD).

18.1 Regulator Operation

A shunt regulator generates a specific supply voltage by creating a voltage drop across a pass resistor RSER. The voltage at the VDD pin of the microcontroller is monitored and compared to an internal voltage reference. The current through the resistor is then adjusted, based on the result of the comparison, to produce a voltage drop equal to the difference between the supply voltage VUNREG and the VDD of the microcontroller. See Figure 18-1 for voltage regulator schematic.

FIGURE 18-1: SHUNT REGULATOR



An external current limiting resistor, RSER, located between the unregulated supply, VUNREG, and the VDD pin, drops the difference in voltage between VUNREG and VDD. RSER must be between RMAX and RMIN as defined by Equation 18-1.

EQUATION 18-1: RSER LIMITING RESISTOR

$$R_{MAX} = \frac{(V_{UMIN} - 5V)}{1.05 \cdot (4\text{ MA} + I_{LOAD})}$$

$$R_{MIN} = \frac{(V_{UMAX} - 5V)}{0.95 \cdot (50\text{ MA})}$$

Where:

RMAX = maximum value of RSER (ohms)

RMIN = minimum value of RSER (ohms)

VUMIN = minimum value of VUNREG

VUMAX = maximum value of VUNREG

VDD = regulated voltage (5V nominal)

ILOAD = maximum expected load current in mA including I/O pin currents and external circuits connected to VDD.

1.05 = compensation for +5% tolerance of RSER

0.95 = compensation for -5% tolerance of RSER

18.2 Regulator Considerations

The supply voltage VUNREG and load current are not constant. Therefore, the current range of the regulator is limited. Selecting a value for RSER must take these three factors into consideration.

Since the regulator uses the band gap voltage as the regulated voltage reference, this voltage reference is permanently enabled in the PIC12HV752 devices.

The shunt regulator will still consume current when below operating voltage range for the shunt regulator.

18.3 Design Considerations

For more information on using the shunt regulator and managing current load, see Application Note AN1035, "Designing with HV Microcontrollers" (DS01035).

PIC12F752/HV752

NOTES:

19.0 DEVELOPMENT SUPPORT

The PIC® microcontrollers and dsPIC® digital signal controllers are supported with a full range of software and hardware development tools:

- Integrated Development Environment
 - MPLAB® IDE Software
- Compilers/Assemblers/Linkers
 - MPLAB C Compiler for Various Device Families
 - HI-TECH C for Various Device Families
 - MPASM™ Assembler
 - MPLINK™ Object Linker/
MPLIB™ Object Librarian
 - MPLAB Assembler/Linker/Librarian for Various Device Families
- Simulators
 - MPLAB SIM Software Simulator
- Emulators
 - MPLAB REAL ICE™ In-Circuit Emulator
- In-Circuit Debuggers
 - MPLAB ICD 3
 - PICKit™ 3 Debug Express
- Device Programmers
 - PICKit™ 2 Programmer
 - MPLAB PM3 Device Programmer
- Low-Cost Demonstration/Development Boards, Evaluation Kits, and Starter Kits

19.1 MPLAB Integrated Development Environment Software

The MPLAB IDE software brings an ease of software development previously unseen in the 8/16/32-bit microcontroller market. The MPLAB IDE is a Windows® operating system-based application that contains:

- A single graphical interface to all debugging tools
 - Simulator
 - Programmer (sold separately)
 - In-Circuit Emulator (sold separately)
 - In-Circuit Debugger (sold separately)
- A full-featured editor with color-coded context
- A multiple project manager
- Customizable data windows with direct edit of contents
- High-level source code debugging
- Mouse over variable inspection
- Drag and drop variables from source to watch windows
- Extensive on-line help
- Integration of select third party tools, such as IAR C Compilers

The MPLAB IDE allows you to:

- Edit your source files (either C or assembly)
- One-touch compile or assemble, and download to emulator and simulator tools (automatically updates all project information)
- Debug using:
 - Source files (C or assembly)
 - Mixed C and assembly
 - Machine code

MPLAB IDE supports multiple debugging tools in a single development paradigm, from the cost-effective simulators, through low-cost in-circuit debuggers, to full-featured emulators. This eliminates the learning curve when upgrading to tools with increased flexibility and power.

PIC12F752/HV752

19.2 MPLAB C Compilers for Various Device Families

The MPLAB C Compiler code development systems are complete ANSI C compilers for Microchip's PIC18, PIC24 and PIC32 families of microcontrollers and the dsPIC30 and dsPIC33 families of digital signal controllers. These compilers provide powerful integration capabilities, superior code optimization and ease of use.

For easy source level debugging, the compilers provide symbol information that is optimized to the MPLAB IDE debugger.

19.3 HI-TECH C for Various Device Families

The HI-TECH C Compiler code development systems are complete ANSI C compilers for Microchip's PIC family of microcontrollers and the dsPIC family of digital signal controllers. These compilers provide powerful integration capabilities, omniscient code generation and ease of use.

For easy source level debugging, the compilers provide symbol information that is optimized to the MPLAB IDE debugger.

The compilers include a macro assembler, linker, pre-processor, and one-step driver, and can run on multiple platforms.

19.4 MPASM Assembler

The MPASM Assembler is a full-featured, universal macro assembler for PIC10/12/16/18 MCUs.

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, absolute LST files that contain source lines and generated machine code and COFF files 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

19.5 MPLINK Object Linker/ MPLIB Object Librarian

The MPLINK Object Linker combines relocatable objects created by the MPASM Assembler and the MPLAB C18 C Compiler. It can link relocatable objects from precompiled libraries, using directives from a linker script.

The MPLIB Object Librarian manages the creation and modification of library files of precompiled code. When a routine from a library is called from a 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 object linker/library features include:

- Efficient linking of single libraries instead of many smaller files
- Enhanced code maintainability by grouping related modules together
- Flexible creation of libraries with easy module listing, replacement, deletion and extraction

19.6 MPLAB Assembler, Linker and Librarian for Various Device Families

MPLAB Assembler produces relocatable machine code from symbolic assembly language for PIC24, PIC32 and dsPIC devices. MPLAB C Compiler uses the assembler to produce its object file. The assembler generates relocatable object files that can then be archived or linked with other relocatable object files and archives to create an executable file. Notable features of the assembler include:

- Support for the entire device instruction set
- Support for fixed-point and floating-point data
- Command line interface
- Rich directive set
- Flexible macro language
- MPLAB IDE compatibility

19.7 MPLAB SIM Software Simulator

The MPLAB SIM Software Simulator allows code development in a PC-hosted environment by simulating the PIC MCUs and dsPIC® DSCs on an instruction level. On any given instruction, the data areas can be examined or modified and stimuli can be applied from a comprehensive stimulus controller. Registers can be logged to files for further run-time analysis. The trace buffer and logic analyzer display extend the power of the simulator to record and track program execution, actions on I/O, most peripherals and internal registers.

The MPLAB SIM Software Simulator fully supports symbolic debugging using the MPLAB C Compilers, and the MPASM and MPLAB Assemblers. The software simulator offers the flexibility to develop and debug code outside of the hardware laboratory environment, making it an excellent, economical software development tool.

19.8 MPLAB REAL ICE In-Circuit Emulator System

MPLAB REAL ICE In-Circuit Emulator System is Microchip's next generation high-speed emulator for Microchip Flash DSC and MCU devices. It debugs and programs PIC® Flash MCUs and dsPIC® Flash DSCs with the easy-to-use, powerful graphical user interface of the MPLAB Integrated Development Environment (IDE), included with each kit.

The emulator is connected to the design engineer's PC using a high-speed USB 2.0 interface and is connected to the target with either a connector compatible with in-circuit debugger systems (RJ11) or with the new high-speed, noise tolerant, Low-Voltage Differential Signal (LVDS) interconnection (CAT5).

The emulator is field upgradable through future firmware downloads in MPLAB IDE. In upcoming releases of MPLAB IDE, new devices will be supported, and new features will be added. MPLAB REAL ICE offers significant advantages over competitive emulators including low-cost, full-speed emulation, run-time variable watches, trace analysis, complex breakpoints, a ruggedized probe interface and long (up to three meters) interconnection cables.

19.9 MPLAB ICD 3 In-Circuit Debugger System

MPLAB ICD 3 In-Circuit Debugger System is Microchip's most cost effective high-speed hardware debugger/programmer for Microchip Flash Digital Signal Controller (DSC) and microcontroller (MCU) devices. It debugs and programs PIC® Flash microcontrollers and dsPIC® DSCs with the powerful, yet easy-to-use graphical user interface of MPLAB Integrated Development Environment (IDE).

The MPLAB ICD 3 In-Circuit Debugger probe is connected to the design engineer's PC using a high-speed USB 2.0 interface and is connected to the target with a connector compatible with the MPLAB ICD 2 or MPLAB REAL ICE systems (RJ-11). MPLAB ICD 3 supports all MPLAB ICD 2 headers.

19.10 PICkit 3 In-Circuit Debugger/Programmer and PICkit 3 Debug Express

The MPLAB PICkit 3 allows debugging and programming of PIC® and dsPIC® Flash microcontrollers at a most affordable price point using the powerful graphical user interface of the MPLAB Integrated Development Environment (IDE). The MPLAB PICkit 3 is connected to the design engineer's PC using a full speed USB interface and can be connected to the target via an Microchip debug (RJ-11) connector (compatible with MPLAB ICD 3 and MPLAB REAL ICE). The connector uses two device I/O pins and the reset line to implement in-circuit debugging and In-Circuit Serial Programming™.

The PICkit 3 Debug Express include the PICkit 3, demo board and microcontroller, hookup cables and CDROM with user's guide, lessons, tutorial, compiler and MPLAB IDE software.

PIC12F752/HV752

19.11 PICkit 2 Development Programmer/Debugger and PICkit 2 Debug Express

The PICkit™ 2 Development Programmer/Debugger is a low-cost development tool with an easy to use interface for programming and debugging Microchip's Flash families of microcontrollers. The full featured Windows® programming interface supports baseline (PIC10F, PIC12F5xx, PIC16F5xx), midrange (PIC12F6xx, PIC16F), PIC18F, PIC24, dsPIC30, dsPIC33, and PIC32 families of 8-bit, 16-bit, and 32-bit microcontrollers, and many Microchip Serial EEPROM products. With Microchip's powerful MPLAB Integrated Development Environment (IDE) the PICkit™ 2 enables in-circuit debugging on most PIC® microcontrollers. In-Circuit-Debugging runs, halts and single steps the program while the PIC microcontroller is embedded in the application. When halted at a breakpoint, the file registers can be examined and modified.

The PICkit 2 Debug Express include the PICkit 2, demo board and microcontroller, hookup cables and CDROM with user's guide, lessons, tutorial, compiler and MPLAB IDE software.

19.12 MPLAB PM3 Device Programmer

The MPLAB PM3 Device Programmer is a universal, CE compliant device programmer with programmable voltage verification at VDDMIN and VDDMAX for maximum reliability. It features a large LCD display (128 x 64) for menus and error messages and a modular, detachable socket assembly to support various package types. The ICSP™ cable assembly is included as a standard item. In Stand-Alone mode, the MPLAB PM3 Device Programmer can read, verify and program PIC devices without a PC connection. It can also set code protection in this mode. The MPLAB PM3 connects to the host PC via an RS-232 or USB cable. The MPLAB PM3 has high-speed communications and optimized algorithms for quick programming of large memory devices and incorporates an MMC card for file storage and data applications.

19.13 Demonstration/Development Boards, Evaluation Kits, and Starter Kits

A wide variety of demonstration, development and evaluation boards for various PIC MCUs and dsPIC DSCs allows quick application development on fully functional systems. Most boards include prototyping areas for adding custom circuitry and provide application firmware and source code for examination and modification.

The boards support a variety of features, including LEDs, temperature sensors, switches, speakers, RS-232 interfaces, LCD displays, potentiometers and additional EEPROM memory.

The demonstration and development boards can be used in teaching environments, for prototyping custom circuits and for learning about various microcontroller applications.

In addition to the PICDEM™ and dsPICDEM™ demonstration/development board series of circuits, Microchip has a line of evaluation kits and demonstration software for analog filter design, KEELOQ® security ICs, CAN, IrDA®, PowerSmart battery management, SEEVAL® evaluation system, Sigma-Delta ADC, flow rate sensing, plus many more.

Also available are starter kits that contain everything needed to experience the specified device. This usually includes a single application and debug capability, all on one board.

Check the Microchip web page (www.microchip.com) for the complete list of demonstration, development and evaluation kits.

20.0 ELECTRICAL SPECIFICATIONS

Absolute Maximum Ratings^(†)

| | |
|--|-----------------------|
| Ambient temperature under bias | -40° to +125°C |
| Storage temperature | -65°C to +150°C |
| Voltage on VDD with respect to VSS | -0.3V to +6.5V |
| Voltage on $\overline{\text{MCLR}}$ with respect to VSS | -0.3V to +13.5V |
| Voltage on all other pins with respect to VSS | -0.3V to (VDD + 0.3V) |
| Total power dissipation ⁽¹⁾ | 800 mW |
| Maximum current out of VSS pin | 95 mA |
| Maximum current into VDD pin | 95 mA |
| Input clamp current, I _{IK} (V _I < 0 or V _I > VDD) | ± 20 mA |
| Output clamp current, I _{OK} (V _O < 0 or V _O > VDD) | ± 20 mA |
| Maximum output current sunk by any I/O pin | 25 mA |
| Maximum output current sourced by any I/O pin | 25 mA |
| Maximum current sunk by PORTA | 90 mA |
| Maximum current sourced PORTA | 90 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 “Absolute 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 above maximum rating conditions for extended periods may affect device reliability.

PIC12F752/HV752

FIGURE 20-1: PIC12F752 VOLTAGE-FREQUENCY GRAPH, $-40^{\circ}\text{C} \leq T_A \leq +125^{\circ}\text{C}$

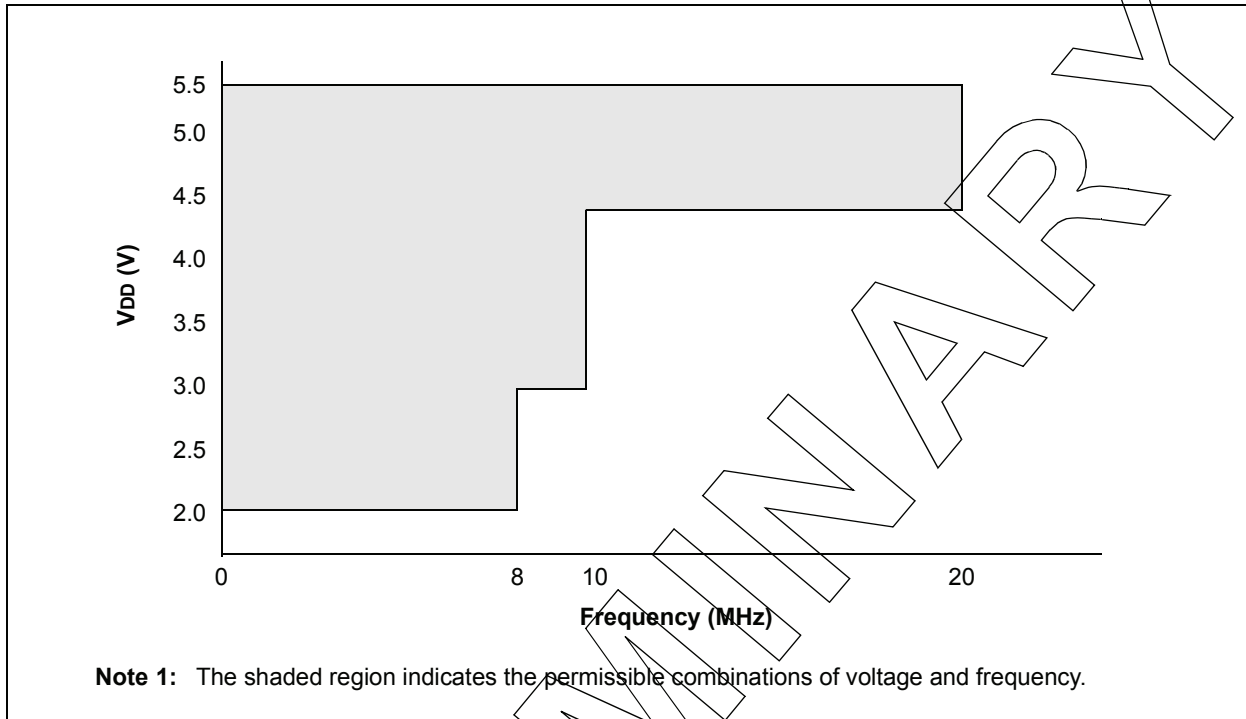


FIGURE 20-2: PIC12HV752 VOLTAGE-FREQUENCY GRAPH, $-40^{\circ}\text{C} \leq T_A \leq +125^{\circ}\text{C}$



PIC12F752/HV752

20.1 DC Characteristics: PIC12F752/HV752-I (Industrial) PIC12F752/HV752-E (Extended)

| DC CHARACTERISTICS | | | Standard Operating Conditions (unless otherwise stated) | | | | |
|-----------------------|------|---|---|------|--------------------|-------|---|
| | | | Operating temperature $-40^{\circ}\text{C} \leq T_A \leq +85^{\circ}\text{C}$ for industrial $-40^{\circ}\text{C} \leq T_A \leq +125^{\circ}\text{C}$ for extended | | | | |
| Param No. | Sym | Characteristic | Min | Typ† | Max | Units | Conditions |
| Supply Voltage | | | | | | | |
| D001 | VDD | PIC12F752 | 2.0 | — | 5.5 | V | Fosc \leq 4 MHz |
| D001 | | PIC12HV752 | 2.0 | — | 5.0 ⁽²⁾ | V | Fosc \leq 4 MHz |
| D001B | VDD | PIC12F752 | 2.0 | — | 5.5 | V | Fosc \leq 8 MHz |
| D001B | | PIC12HV752 | 2.0 | — | 5.0 ⁽²⁾ | V | Fosc \leq 8 MHz |
| D001C | VDD | PIC12F752 | 3.0 | — | 5.5 | V | Fosc \leq 10 MHz |
| D001C | | PIC12HV752 | 3.0 | — | 5.0 ⁽²⁾ | V | Fosc \leq 10 MHz |
| D001D | VDD | PIC12F752 | 4.5 | — | 5.5 | V | Fosc \leq 20 MHz |
| D001D | | PIC12HV752 | 4.5 | — | 5.0 ⁽²⁾ | V | Fosc \leq 20 MHz |
| D002* | VDR | RAM Data Retention Voltage⁽¹⁾ | 1.5 | — | — | V | Device in Sleep mode |
| D003 | VPOR | VDD Start Voltage to ensure internal Power-on Reset signal | — | VSS | — | V | See Section TABLE 17-1: “Time-out in Various Situations” for details. |
| D004* | SVDD | VDD Rise Rate to ensure internal Power-on Reset signal | 0.05 | — | — | V/ms | See Section TABLE 17-1: “Time-out in Various Situations” for details. |

* These parameters are characterized but not tested.

† Data in “Typ” column is at 5.0V, 25°C unless otherwise stated. These parameters are for design guidance only and are not tested.

Note 1: This is the limit to which VDD can be lowered in Sleep mode without losing RAM data.

Note 2: On the PIC12HV752, VDD is regulated by a Shunt Regulator and is dependent on series resistor (connected between the unregulated supply voltage and the VDD pin) to limit the current to 50 mA. See Section 18.0 “Shunt Regulator (PIC12HV752 Only)” for design requirements.

PIC12F752/HV752

20.2 DC Characteristics: PIC12F752-I (Industrial) PIC12F752-E (Extended)

| DC CHARACTERISTICS | | Standard Operating Conditions (unless otherwise stated) Operating temperature $-40^{\circ}\text{C} \leq T_A \leq +85^{\circ}\text{C}$ for industrial $-40^{\circ}\text{C} \leq T_A \leq +125^{\circ}\text{C}$ for extended | | | | | |
|--------------------|---|--|-------|-----|-------|------------|-------------------------------------|
| Param No. | Device Characteristics | Min | Typ† | Max | Units | Conditions | |
| | | | | | | VDD | Note |
| D010 | Supply Current (IDD) ^(1, 2) PIC12F752 | — | 8 | TBD | μA | 2.0 | Fosc = 31 kHz LFINTOSC mode |
| | | — | 16 | TBD | μA | 3.0 | |
| | | — | 31 | TBD | μA | 5.0 | |
| D011* | | — | 140 | TBD | μA | 2.0 | Fosc = 1 MHz HFINTOSC mode |
| | | — | 215 | TBD | μA | 3.0 | |
| | | — | 360 | TBD | μA | 5.0 | |
| D012* | | — | 185 | TBD | μA | 2.0 | Fosc = 4 MHz EC Oscillator mode |
| | | — | 325 | TBD | μA | 3.0 | |
| | | — | 665 | TBD | μA | 5.0 | |
| D013* | | — | 270 | TBD | μA | 2.0 | Fosc = 4 MHz HFINTOSC mode |
| | | — | 390 | TBD | μA | 3.0 | |
| | | — | 680 | TBD | μA | 5.0 | |
| D014 | | — | 0.395 | TBD | mA | 2.0 | Fosc = 8 MHz HFINTOSC mode |
| | | — | 0.620 | TBD | mA | 3.0 | |
| | | — | 1.2 | TBD | mA | 5.0 | |
| D015 | | — | 2.6 | TBD | mA | 4.5 | Fosc = 20 MHz EC Oscillator mode |
| | | — | 2.8 | TBD | mA | 5.0 | |
| D016 | | — | 65 | — | μA | 2.0 | Fosc = 1 MHz EC Oscillator mode |
| | | — | 120 | — | μA | 3.0 | |
| | | — | 250 | — | μA | 5.0 | |

Legend: TBD = To Be Determined

* These parameters are characterized but not tested.

† Data in "Typ" column is at 5.0V, 25°C unless otherwise stated. These parameters are for design guidance only and are not tested.

Note 1: The test conditions for all IDD measurements in active operation mode are: CLKIN = external square wave, from rail-to-rail, all I/O pins tri-stated, pulled to VDD; MCLR = VDD; WDT disabled.

The supply current is mainly a function of the operating voltage and frequency. Other factors, such as I/O pin loading and switching rate, oscillator type, internal code execution pattern and temperature, also have an impact on the current consumption.

20.3 DC Characteristics: PIC12HV752-I (Industrial) PIC12HV752-E (Extended)

| DC CHARACTERISTICS | | Standard Operating Conditions (unless otherwise stated) Operating temperature $-40^{\circ}\text{C} \leq T_A \leq +85^{\circ}\text{C}$ for industrial $-40^{\circ}\text{C} \leq T_A \leq +125^{\circ}\text{C}$ for extended | | | | | |
|--------------------|--|--|-------|-----|-------|------------|--|
| Param No. | Device Characteristics | Min | Typ† | Max | Units | Conditions | |
| | | | | | | VDD | Note |
| D010 | Supply Current (IDD) ^(1, 2) PIC12HV752 | — | 20 | TBD | μA | 2.0 | Fosc = 31 kHz LFINTOSC mode |
| | | — | 40 | TBD | μA | 3.0 | |
| | | — | 65 | TBD | μA | 4.5 | |
| D011* | | — | 215 | TBD | μA | 2.0 | Fosc = 1 MHz HFINTOSC Oscillator mode |
| | | — | 375 | TBD | μA | 3.0 | |
| | | — | 570 | TBD | μA | 4.5 | |
| D012* | | — | 335 | TBD | μA | 2.0 | Fosc = 4 MHz HFINTOSC mode |
| | | — | 535 | TBD | μA | 3.0 | |
| | | — | 795 | TBD | μA | 4.5 | |
| D013 | | — | 0.530 | TBD | mA | 2.0 | Fosc = 8 MHz HFINTOSC mode |
| | | — | 0.830 | TBD | mA | 3.0 | |
| | | — | 1.35 | TBD | mA | 4.5 | |
| D014 | | — | 3.8 | TBD | mA | 4.5 | Fosc = 20 MHz EC Oscillator mode |
| D015 | | — | 130 | — | μA | 2.0 | Fosc = 1 MHz EC Oscillator mode |
| | | — | 245 | — | μA | 3.0 | |
| | | — | 360 | — | μA | 5.0 | |
| D016 | | — | 250 | — | μA | 2.0 | Fosc = 4 MHz EC Oscillator mode |
| | | — | 440 | — | μA | 3.0 | |
| | | — | 660 | — | μA | 5.0 | |

Legend: TBD = To Be Determined

* These parameters are characterized but not tested.

† Data in "Typ" column is at 4.5V, 25°C unless otherwise stated. These parameters are for design guidance only and are not tested.

Note 1: The test conditions for all IDD measurements in active operation mode are: CLKIN = external square wave, from rail-to-rail; all I/O pins tri-stated, pulled to VDD; MCLR = VDD; WDT disabled.

2: The supply current is mainly a function of the operating voltage and frequency. Other factors, such as I/O pin loading and switching rate, oscillator type, internal code execution pattern and temperature, also have an impact on the current consumption.

3: For RC oscillator configurations, current through REXT is not included. The current through the resistor can be extended by the formula $I_R = V_{DD}/2R_{EXT}$ (mA) with REXT in kΩ.

PIC12F752/HV752

20.4 DC Characteristics: PIC12F752 - I (Industrial)

| DC CHARACTERISTICS | | Standard Operating Conditions (unless otherwise stated) Operating temperature $-40^{\circ}\text{C} \leq T_A \leq +85^{\circ}\text{C}$ for industrial | | | | | Conditions | |
|--------------------|--|---|------|-----|---------------|-----|---|--|
| Param No. | Device Characteristics | Min | Typ† | Max | Units | VDD | Note | |
| | | | | | | | | |
| D020 | Power-down Base Current (IPD) ⁽²⁾ | — | 0.01 | TBD | μA | 2.0 | WDT, BOR, Comparator, VREF and T1OSC disabled | |
| | | — | 0.3 | TBD | μA | 3.0 | | |
| | PIC12F752 | — | 0.5 | TBD | μA | 5.0 | $-40^{\circ}\text{C} \leq T_A \leq +25^{\circ}\text{C}$ for industrial | |
| | | — | 150 | TBD | nA | 3.0 | | |
| D021 | | — | 3.0 | TBD | μA | 2.0 | WDT Current ⁽¹⁾ | |
| | | — | 4.0 | TBD | μA | 3.0 | | |
| | | — | 7.0 | TBD | μA | 5.0 | | |
| D022 | | — | 5.0 | TBD | μA | 3.0 | BOR Current ⁽¹⁾ | |
| | | — | 6.0 | TBD | μA | 5.0 | | |
| D023 | | — | 362 | TBD | μA | 2.0 | CxSP = 1, Comparator Current ⁽¹⁾ , single comparator enabled | |
| | | — | 418 | TBD | μA | 3.0 | | |
| | | — | 500 | TBD | μA | 5.0 | | |
| D024 | | — | 96 | TBD | μA | 2.0 | CxSP = 0, Comparator Current ⁽¹⁾ , single comparator enabled | |
| | | — | 112 | TBD | μA | 3.0 | | |
| | | — | 132 | TBD | μA | 5.0 | | |
| D025 | | — | 0.3 | TBD | μA | 3.0 | A/D Current ⁽¹⁾ , no conversion in progress | |
| | | — | 0.36 | TBD | μA | 5.0 | | |
| D026 | | — | 0.2 | TBD | μA | 3.0 | DAC Current ⁽¹⁾ | |
| | | — | 0.4 | TBD | μA | 5.0 | | |
| D027 | | — | 59 | TBD | μA | 3.0 | FVR Current ⁽¹⁾ , FVRBUFEN = 1, REFOUT buffer enabled | |
| | | — | 98 | TBD | μA | 5.0 | | |

Legend: TBD = To Be Determined

* These parameters are characterized but not tested.

† Data in "Typ" column is at 5.0V, 25°C unless otherwise stated. These parameters are for design guidance only and are not tested.

Note 1: The peripheral current is the sum of the base IDD or IPD and the additional current consumed when this peripheral is enabled. The peripheral Δ current can be determined by subtracting the base IDD or IPD current from this limit. Max values should be used when calculating total current consumption.

2: 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 high-impedance state and tied to VDD.

20.5 DC Characteristics: PIC12F752 - E (Extended)

| DC CHARACTERISTICS | | Standard Operating Conditions (unless otherwise stated) Operating temperature $-40^{\circ}\text{C} \leq T_A \leq +125^{\circ}\text{C}$ for extended | | | | | Conditions | |
|--------------------|---|--|------|-----|-------|-----------------|---|--|
| Param No. | Device Characteristics | Min | Typ† | Max | Units | Conditions | | |
| | | | | | | V _{DD} | Note | |
| D020E | Power-down Base Current (IPD) ⁽²⁾ PIC12F752 | — | 0.1 | TBD | μA | 2.0 | WDT, BOR, Comparator, DAC and FVR disabled | |
| | | — | 0.3 | TBD | μA | 3.0 | | |
| | | — | 0.5 | TBD | μA | 5.0 | | |
| D021E | | — | 3.0 | TBD | μA | 2.0 | WDT Current ⁽¹⁾ | |
| | | — | 4.0 | TBD | μA | 3.0 | | |
| | | — | 7.0 | TBD | μA | 5.0 | | |
| D022E | | — | 5.0 | TBD | μA | 3.0 | BOR Current ⁽¹⁾ | |
| | | — | 6.0 | TBD | μA | 5.0 | | |
| D023E | | — | 362 | TBD | μA | 2.0 | CxSP = 1, Comparator Current ⁽¹⁾ , single comparator enabled | |
| | | — | 418 | TBD | μA | 3.0 | | |
| | | — | 500 | TBD | μA | 5.0 | | |
| D024E | | — | 96 | TBD | μA | 2.0 | CxSP = 0, Comparator Current ⁽¹⁾ , single comparator enabled | |
| | | — | 112 | TBD | μA | 3.0 | | |
| | | — | 132 | TBD | μA | 5.0 | | |
| D025E | | — | 03 | TBD | μA | 3.0 | A/D Current ⁽¹⁾ , no conversion in progress | |
| | | — | 0.36 | TBD | μA | 5.0 | | |
| D026E | | — | 0.2 | TBD | μA | 3.0 | DAC Current ^(1,3) | |
| | | — | 0.4 | TBD | μA | 5.0 | | |
| D027E | | — | 59 | TBD | μA | 3.0 | FVR Current ⁽¹⁾ , FVRBUFEN = 1, REFOUT buffer enabled | |
| | | — | 98 | TBD | μA | 5.0 | | |

Legend: TBD = To Be Determined

* These parameters are characterized but not tested.

† Data in "Typ" column is at 5.0V, 25°C unless otherwise stated. These parameters are for design guidance only and are not tested.

Note 1: The peripheral current is the sum of the base I_{DD} or I_{PD} and the additional current consumed when this peripheral is enabled. The peripheral Δ current can be determined by subtracting the base I_{DD} or I_{PD} current from this limit. Max values should be used when calculating total current consumption.

2: 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 high-impedance state and tied to V_{DD}.

3: Both or one input reference are in high z-state.

PIC12F752/HV752

20.6 DC Characteristics: PIC12HV752 - I (Industrial)

| DC CHARACTERISTICS | | Standard Operating Conditions (unless otherwise stated) Operating temperature $-40^{\circ}\text{C} \leq T_A \leq +85^{\circ}\text{C}$ for industrial | | | | | |
|--------------------|--|---|------|-----|-------|------------|---|
| Param No. | Device Characteristics | Min | Typ† | Max | Units | Conditions | |
| | | | | | | VDD | Note |
| D020 | Power-down Base Current (IPD) ^(2,3) PIC12HV752 | — | 135 | TBD | μA | 2.0 | WDT, BOR, Comparator, DAC and FVR disabled |
| | | — | 210 | TBD | μA | 3.0 | |
| | | — | 260 | TBD | μA | 4.5 | |
| D021 | | — | 135 | TBD | μA | 2.0 | WDT Current ⁽¹⁾ |
| | | — | 210 | TBD | μA | 3.0 | |
| | | — | 265 | TBD | μA | 4.5 | |
| D022 | | — | 215 | TBD | μA | 3.0 | BOR Current ⁽¹⁾ |
| | | — | 265 | TBD | μA | 4.5 | |
| D023 | | — | 362 | TBD | μA | 2.0 | CxSP = 1, Comparator Current ⁽¹⁾ , single comparator enabled |
| | | — | 418 | TBD | μA | 3.0 | |
| | | — | 450 | TBD | μA | 4.5 | |
| D024 | | — | 90 | TBD | μA | 2.0 | CxSP = 0, Comparator Current ⁽¹⁾ , single comparator enabled |
| | | — | 107 | TBD | μA | 3.0 | |
| | | — | 127 | TBD | μA | 4.5 | |
| D025 | | — | 210 | TBD | μA | 3.0 | A/D Current ⁽¹⁾ , no conversion in progress |
| | | — | 260 | TBD | μA | 4.5 | |
| D026 | | — | 215 | TBD | μA | 3.0 | DAC Current ⁽¹⁾ |
| | | — | 265 | TBD | μA | 4.5 | |
| D027 | | — | 485 | TBD | μA | 3.0 | FVR Current ⁽¹⁾ , FVRBUFEN = 1, REFOUT buffer enabled |
| | | — | 615 | TBD | μA | 4.5 | |

Legend: TBD = To Be Determined

* These parameters are characterized but not tested.

† Data in "Typ" column is at 4.5V, 25°C unless otherwise stated. These parameters are for design guidance only and are not tested.

Note 1: The peripheral current is the sum of the base IDD or IPD and the additional current consumed when this peripheral is enabled. The peripheral Δ current can be determined by subtracting the base IDD or IPD current from this limit. Max values should be used when calculating total current consumption.

2: 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 high-impedance state and tied to VDD.

3: Shunt regulator is always on and always draws operating current.

20.7 DC Characteristics: PIC12HV752-E (Extended)

| DC CHARACTERISTICS | | Standard Operating Conditions (unless otherwise stated) Operating temperature $-40^{\circ}\text{C} \leq T_A \leq +125^{\circ}\text{C}$ for extended | | | | | |
|--------------------|--|--|------|-----|---------------|------------|---|
| Param No. | Device Characteristics | Min | Typ† | Max | Units | Conditions | |
| | | | | | | VDD | Note |
| D020E | Power-down Base Current (IPD) ^(2,3) PIC12HV752 | — | 135 | TBD | μA | 2.0 | WDT, BOR, Comparator, DAC and FVR disabled |
| | | — | 210 | TBD | μA | 3.0 | |
| | | — | 260 | TBD | μA | 4.5 | |
| D021E | | — | 135 | TBD | μA | 2.0 | WDT Current ⁽¹⁾ |
| | | — | 210 | TBD | μA | 3.0 | |
| | | — | 265 | TBD | μA | 4.5 | |
| D022E | | — | 215 | TBD | μA | 3.0 | BOR Current ⁽¹⁾ |
| | | — | 265 | TBD | μA | 4.5 | |
| D023E | | — | 185 | TBD | μA | 2.0 | CxSP = 1, Comparator Current ⁽¹⁾ , single comparator enabled |
| | | — | 265 | TBD | μA | 3.0 | |
| | | — | 320 | TBD | μA | 4.5 | |
| D024E | | — | 90 | TBD | μA | 2.0 | CxSP = 0, Comparator Current ⁽¹⁾ , single comparator enabled |
| | | — | 107 | TBD | μA | 3.0 | |
| | | — | 127 | TBD | μA | 4.5 | |
| D025E | | — | 210 | TBD | μA | 3.0 | A/D Current ⁽¹⁾ , no conversion in progress |
| | | — | 260 | TBD | μA | 4.5 | |
| D026E | | — | 215 | TBD | μA | 3.0 | DAC Current ⁽¹⁾ |
| | | — | 265 | TBD | μA | 4.5 | |
| D027E | | — | 485 | TBD | μA | 3.0 | FVR Current ⁽¹⁾ , FVRBUFEN = 1, REFOUT buffer enabled |
| | | — | 615 | TBD | μA | 4.5 | |

Legend: TBD = To Be Determined

* These parameters are characterized but not tested.

† Data in "Typ" column is at 4.5V, 25°C unless otherwise stated. These parameters are for design guidance only and are not tested.

Note 1: The peripheral current is the sum of the base IDD or IPD and the additional current consumed when this peripheral is enabled. The peripheral Δ current can be determined by subtracting the base IDD or IPD current from this limit. Max values should be used when calculating total current consumption.

2: 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 high-impedance state and tied to VDD.

3: Shunt regulator is always on and always draws operating current.

PIC12F752/HV752

20.8 DC Characteristics: PIC12F752/HV752-I (Industrial) PIC12F752/HV752-E (Extended)

| DC CHARACTERISTICS | | | Standard Operating Conditions (unless otherwise stated) | | | | |
|--------------------|------------------|---|---|-------|----------------------|-------|---|
| | | | Operating temperature -40°C ≤ TA ≤ +85°C for industrial -40°C ≤ TA ≤ +125°C for extended | | | | |
| Param No. | Sym | Characteristic | Min | Typ† | Max | Units | Conditions |
| D030 | V _{IL} | Input Low Voltage I/O port: with TTL buffer | V _{SS} | — | 0.8 | V | 4.5V ≤ V _{DD} ≤ 5.5V |
| D030A | | | V _{SS} | — | 0.15 V _{DD} | V | 2.0V ≤ V _{DD} ≤ 4.5V |
| D031 | | with Schmitt Trigger buffer | V _{SS} | — | 0.2 V _{DD} | V | 2.0V ≤ V _{DD} ≤ 5.5V |
| D040 | V _{IH} | Input High Voltage I/O ports: with TTL buffer | 2.0 | — | V _{DD} | V | 4.5V ≤ V _{DD} ≤ 5.5V |
| D040A | | | 0.25 V _{DD} + 0.8 | — | V _{DD} | V | 2.0V ≤ V _{DD} ≤ 4.5V |
| D041 | | with Schmitt Trigger buffer | 0.8 V _{DD} | — | V _{DD} | V | 2.0V ≤ V _{DD} ≤ 5.5V |
| D042 | | MCLR | 0.8 V _{DD} | — | V _{DD} | V | |
| D060 | I _{IL} | Input Leakage Current ^(1,2) I/O ports | — | ± 0.1 | ± 1 | μA | V _{SS} ≤ V _{PIN} ≤ V _{DD} , Pin at high-impedance |
| D061 | | RA3/MCLR ^(2,3) | — | ± 0.7 | ± 5 | μA | V _{SS} ≤ V _{PIN} ≤ V _{DD} |
| D063 | | | — | ± 0.1 | ± 5 | μA | EC Configuration |
| D070* | I _{PUR} | PORTA Weak Pull-up Current ⁽⁴⁾ | 50 | 250 | 400 | μA | V _{DD} = 5.0V, V _{PIN} = V _{SS} |
| D080 | V _{OL} | Output Low Voltage I/O ports (excluding RA0, RA2) | — | — | 0.6 | V | I _{OL} = 7.0 mA, V _{DD} = 4.5V, -40°C to +125°C |
| | | | — | — | 0.6 | V | I _{OL} = 8.5 mA, V _{DD} = 4.5V, -40°C to +85°C |
| | | RA0, RA2 | — | — | 0.6 | V | I _{OL} = 14 mA, V _{DD} = 4.5V, -40°C to +125°C |
| | | | — | — | 0.6 | V | I _{OL} = 17 mA, V _{DD} = 4.5V, -40°C to +85°C |
| D090 | V _{OH} | Output High Voltage I/O ports ⁽⁴⁾ (excluding RA0, RA2) | V _{DD} - 0.7 | — | — | V | I _{OH} = -2.5mA, V _{DD} = 4.5V, -40°C to +125°C |
| | | | V _{DD} - 0.7 | — | — | V | I _{OH} = -3.0 mA, V _{DD} = 4.5V, -40°C to +85°C |
| | | RA0, RA2 | V _{DD} - 0.7 | — | — | V | I _{OH} = -5.0 mA, V _{DD} = 4.5V, -40°C to +125°C |
| | | | V _{DD} - 0.7 | — | — | V | I _{OH} = -6.0 mA, V _{DD} = 4.5V, -40°C to +85°C |

* These parameters are characterized but not tested.

† Data in "Typ" column is at 5.0V, 25°C unless otherwise stated. These parameters are for design guidance only and are not tested.

- Note**
- 1: Negative current is defined as current sourced by the pin.
 - 2: The leakage current on the 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: This specification applies to RA3/MCLR configured as RA3 with the internal weak pull-up disabled.
 - 4: This specification applies to all weak pull-up pins, including the weak pull-up found on RA3/MCLR. When RA3/MCLR is configured as MCLR Reset pin, the weak pull-up is always enabled.

20.8 DC Characteristics: PIC12F752/HV752-I (Industrial) PIC12F752/HV752-E (Extended) (Continued)

| DC CHARACTERISTICS | | | Standard Operating Conditions (unless otherwise stated) | | | | |
|-----------------------------|-------|--|---|------|-----|-------|---|
| | | | Operating temperature -40°C ≤ TA ≤ +85°C for industrial -40°C ≤ TA ≤ +125°C for extended | | | | |
| Param No. | Sym | Characteristic | Min | Typ† | Max | Units | Conditions |
| D101* | CIO | Capacitive Loading Specs on All I/O pins | — | — | 50 | pF | |
| Program Flash Memory | | | | | | | |
| D130 | EP | Cell Endurance | 10K | 100K | — | E/W | -40°C ≤ TA ≤ +85°C |
| D130A | ED | Cell Endurance | 1K | 10K | — | E/W | +85°C ≤ TA ≤ +125°C |
| D131 | VPR | VDD for Read | V _{MIN} | — | 5.5 | V | V _{MIN} = Minimum operating voltage |
| D132 | VPEW | VDD for Bulk Erase/Write | 4.5 | — | 5.5 | V | |
| D132A | VPEW | VDD for Row Erase/Write | V _{MIN} | — | 5.5 | V | |
| D133 | TPEW | Erase/Write cycle time | — | 2 | 2.5 | ms | |
| D134 | TRETD | Characteristic Retention | 40 | — | — | Year | Provided no other specifications are violated |

* These parameters are characterized but not tested.

† Data in "Typ" column is at 5.0V, 25°C unless otherwise stated. These parameters are for design guidance only and are not tested.

Note 1: Negative current is defined as current sourced by the pin.

2: The leakage current on the 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: This specification applies to RA3/MCLR configured as RA3 with the internal weak pull-up disabled.

4: This specification applies to all weak pull-up pins, including the weak pull-up found on RA3/MCLR. When RA3/MCLR is configured as MCLR Reset pin, the weak pull-up is always enabled.

PRELIMINARY

PIC12F752/HV752

20.9 Thermal Considerations

| Standard Operating Conditions (unless otherwise stated) | | | | | |
|--|----------------|--|--------|-------|--|
| Operating temperature $-40^{\circ}\text{C} \leq T_A \leq +125^{\circ}\text{C}$ | | | | | |
| Param No. | Sym | Characteristic | Typ | Units | Conditions |
| TH01 | θ_{JA} | Thermal Resistance Junction to Ambient | 84.6* | C/W | 8-pin PDIP package |
| | | | 149.5* | C/W | 8-pin SOIC package |
| | | | 60* | C/W | 8-pin DFN 3x3mm package |
| TH02 | θ_{JC} | Thermal Resistance Junction to Case | 41.2* | C/W | 8-pin PDIP package |
| | | | 39.9* | C/W | 8-pin SOIC package |
| | | | 9* | C/W | 8-pin DFN 3x3mm package |
| TH03 | T_{DIE} | Die Temperature | 150* | C | |
| TH04 | PD | Power Dissipation | — | W | $PD = P_{INTERNAL} + P_{I/O}$ |
| TH05 | $P_{INTERNAL}$ | Internal Power Dissipation | — | W | $P_{INTERNAL} = I_{DD} \times V_{DD}$ (NOTE 1) |
| TH06 | $P_{I/O}$ | I/O Power Dissipation | — | W | $P_{I/O} = \sum (I_{OL} \times V_{OL}) + \sum (I_{OH} \times (V_{DD} - V_{OH}))$ |
| TH07 | P_{DER} | Derated Power | — | W | $P_{DER} = P_{D_{MAX}} (T_{DIE} - T_A) / \theta_{JA}$ (NOTE 2) |

* These parameters are characterized but not tested.

Note 1: I_{DD} is current to run the chip alone without driving any load on the output pins.

2: T_A = Ambient temperature.

PRELIMINARY

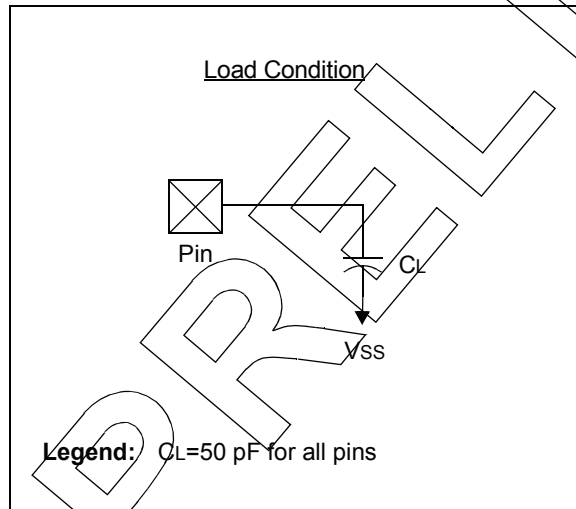
20.10 Timing Parameter Symbology

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

1. TppS2ppS
2. TppS

| | | |
|--|---|---|
| T | F Frequency | T Time |
| Lowercase letters (pp) and their meanings: | | |
| pp | cc CCP1 ck CLKOUT cs \overline{CS} di SDI do SDO dt Data in io I/O Port mc MCLR | osc OSC1 rd \overline{RD} rw \overline{RD} or \overline{WR} sc SCK ss \overline{SS} t0 T0CKI t1 T1CKI wr \overline{WR} |
| Uppercase letters and their meanings: | | |
| S | F Fall H High I Invalid (High-Impedance) L Low | P Period R Rise V Valid Z High-Impedance |

FIGURE 20-3: LOAD CONDITIONS



PIC12F752/HV752

20.11 AC Characteristics: PIC12F752/HV752 (Industrial, Extended)

FIGURE 20-4: CLOCK TIMING

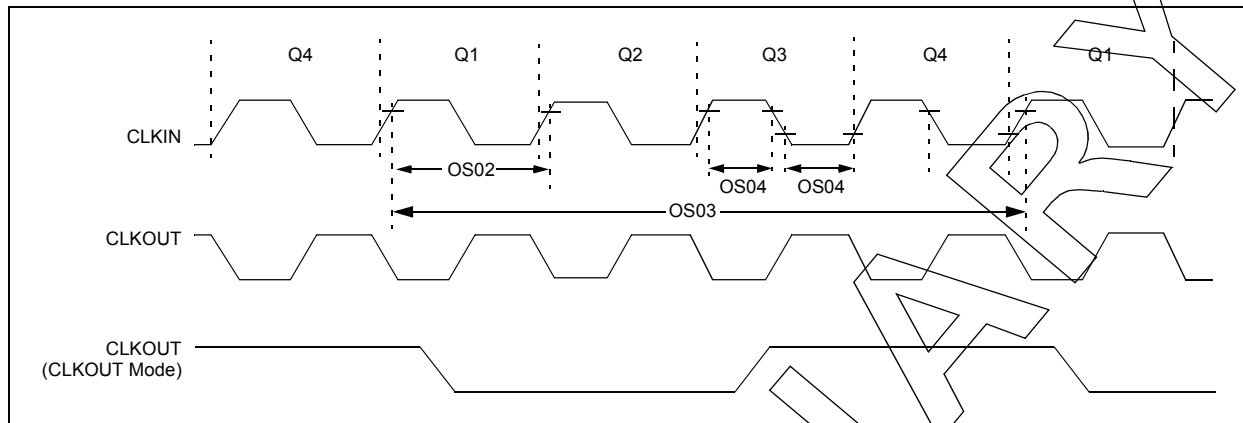


TABLE 20-1: CLOCK OSCILLATOR TIMING REQUIREMENTS

| Standard Operating Conditions (unless otherwise stated) | | | | | | | |
|---|------|--|-----|------|----------|-------|--------------------|
| | | Operating temperature $-40^{\circ}\text{C} \leq T_A \leq +125^{\circ}\text{C}$ | | | | | |
| Param No. | Sym | Characteristic | Min | Typ† | Max | Units | Conditions |
| OS01 | FOSC | External CLKIN Frequency ⁽¹⁾ | DC | — | 20 | MHz | EC Oscillator mode |
| OS02 | TOSC | External CLKIN Period ⁽¹⁾ | 50 | — | ∞ | ns | EC Oscillator mode |
| OS03 | TCY | Instruction Cycle Time ⁽¹⁾ | 200 | TCY | DC | ns | TCY = 4/FOSC |

* These parameters are characterized but not tested.

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

TABLE 20-2: OSCILLATOR PARAMETERS

| Standard Operating Conditions (unless otherwise stated) | | | | | | | | |
|--|---------|---|-----------------|------|------|------|---------------|--|
| Operating Temperature $-40^{\circ}\text{C} \leq T_A \leq +125^{\circ}\text{C}$ | | | | | | | | |
| Param No. | Sym | Characteristic | Freq. Tolerance | Min | Typ† | Max | Units | Conditions |
| OS06 | TWARM | Internal Oscillator Switch when running ⁽³⁾ | — | — | — | 2 | Tosc | Slowest clock |
| OS07 | INTOSC | Internal Calibrated INTOSC Frequency ⁽²⁾ (4 MHz) | $\pm 1\%$ | 3.96 | 4.0 | 4.04 | MHz | $V_{DD} = 3.5\text{V}, T_A = 25^{\circ}\text{C}$ $2.5\text{V} \leq V_{DD} \leq 5.5\text{V}, 0^{\circ}\text{C} \leq T_A \leq +85^{\circ}\text{C}$ $2.0\text{V} \leq V_{DD} \leq 5.5\text{V}, -40^{\circ}\text{C} \leq T_A \leq +85^{\circ}\text{C (Ind.)}, -40^{\circ}\text{C} \leq T_A \leq +125^{\circ}\text{C (Ext.)}$ |
| | | | $\pm 2\%$ | 3.92 | 4.0 | 4.08 | MHz | |
| | | | $\pm 5\%$ | 3.80 | 4.0 | 4.2 | MHz | |
| OS08 | INTOSC | Internal Calibrated INTOSC Frequency ⁽²⁾ (8 MHz) | $\pm 1\%$ | 7.92 | 8.0 | 8.08 | MHz | $V_{DD} = 3.5\text{V}, T_A = 25^{\circ}\text{C}$ $2.5\text{V} \leq V_{DD} \leq 5.5\text{V}, 0^{\circ}\text{C} \leq T_A \leq +85^{\circ}\text{C}$ $2.0\text{V} \leq V_{DD} \leq 5.5\text{V}, -40^{\circ}\text{C} \leq T_A \leq +85^{\circ}\text{C (Ind.)}, -40^{\circ}\text{C} \leq T_A \leq +125^{\circ}\text{C (Ext.)}$ |
| | | | $\pm 2\%$ | 7.84 | 8.0 | 8.16 | MHz | |
| | | | $\pm 5\%$ | 7.60 | 8.0 | 8.40 | MHz | |
| OS10* | TOSC ST | INTOSC Oscillator Wake-up from Sleep Start-up Time | — | TBD | 12 | TBD | μs | $V_{DD} = 2.0\text{V}, -40^{\circ}\text{C to } +85^{\circ}\text{C}$ |
| | | | — | TBD | 7 | TBD | μs | $V_{DD} = 3.0\text{V}, -40^{\circ}\text{C to } +85^{\circ}\text{C}$ |
| | | | — | TBD | 6 | TBD | μs | $V_{DD} = 5.0\text{V}, -40^{\circ}\text{C to } +85^{\circ}\text{C}$ |

Legend: TBD = To Be Determined

* These parameters are characterized but not tested.

† Data in "Typ" column is at 5.0V, 25°C unless otherwise stated. These parameters are for design guidance only and are not tested.

- Note 1:** Instruction cycle period (TCY) equals four times the input oscillator time base period. All specified values are based on characterization data for that particular oscillator type under standard operating conditions with the device executing code. Exceeding these specified limits may result in an unstable oscillator operation and/or higher than expected current consumption. All devices are tested to operate at "min" values with an external clock applied to the CLKIN pin. When an external clock input is used, the "max" cycle time limit is "DC" (no clock) for all devices.
- 2:** To ensure these oscillator frequency tolerances, VDD and VSS must be capacitively decoupled as close to the device as possible. 0.1 μF and 0.01 μF values in parallel are recommended.
- 3:** By design.

PRELIMINARY

PIC12F752/HV752

FIGURE 20-5: CLKOUT AND I/O TIMING

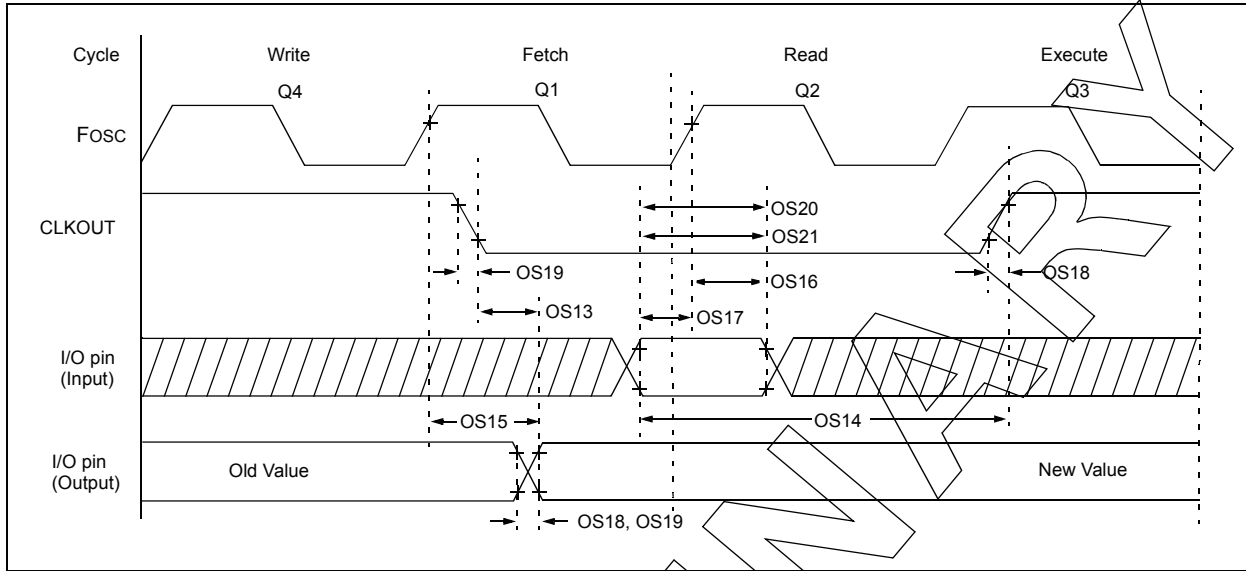


TABLE 20-3: CLKOUT AND I/O TIMING PARAMETERS

| Standard Operating Conditions (unless otherwise stated) Operating Temperature $-40^{\circ}\text{C} \leq T_A \leq +125^{\circ}\text{C}$ | | | | | | | |
|---|----------|---|---------------|----------|----------|-------|--------------------------|
| Param No. | Sym | Characteristic | Min | Typ† | Max | Units | Conditions |
| OS13 | TckL2ioV | CLKOUT \downarrow to Port out valid ⁽¹⁾ | — | — | 20 | ns | |
| OS14 | TioV2ckH | Port input valid before CLKOUT \uparrow ⁽¹⁾ | Tosc + 200 ns | — | — | ns | |
| OS15 | Tosh2ioV | Fosc \uparrow (Q1 cycle) to Port out valid | — | 50 | 70* | ns | VDD = 5.0V |
| OS16 | Tosh2ioI | Fosc \uparrow (Q2 cycle) to Port input invalid (I/O in hold time) | 50 | — | — | ns | VDD = 5.0V |
| OS17 | TioV2osH | Port input valid to Fosc \uparrow (Q2 cycle) (I/O in setup time) | 20 | — | — | ns | |
| OS18 | TioR | Port output rise time ⁽¹⁾ | — | 15 40 | 72 32 | ns | VDD = 2.0V VDD = 5.0V |
| OS19 | TioF | Port output fall time ⁽¹⁾ | — | 28 15 | 55 30 | ns | VDD = 2.0V VDD = 5.0V |
| OS20* | TINP | INT pin input high or low time | 25 | — | — | ns | |
| OS21* | TRAP | PORTA interrupt-on-change new input level time | TCY | — | — | ns | |

These parameters are characterized but not tested.

† Data in "Typ" column is at 5.0V, 25°C unless otherwise stated.

Note 1: Includes CLKOUTEN = 0, CLKOUT function enabled.

FIGURE 20-6: RESET, WATCHDOG TIMER, AND POWER-UP TIMER TIMING

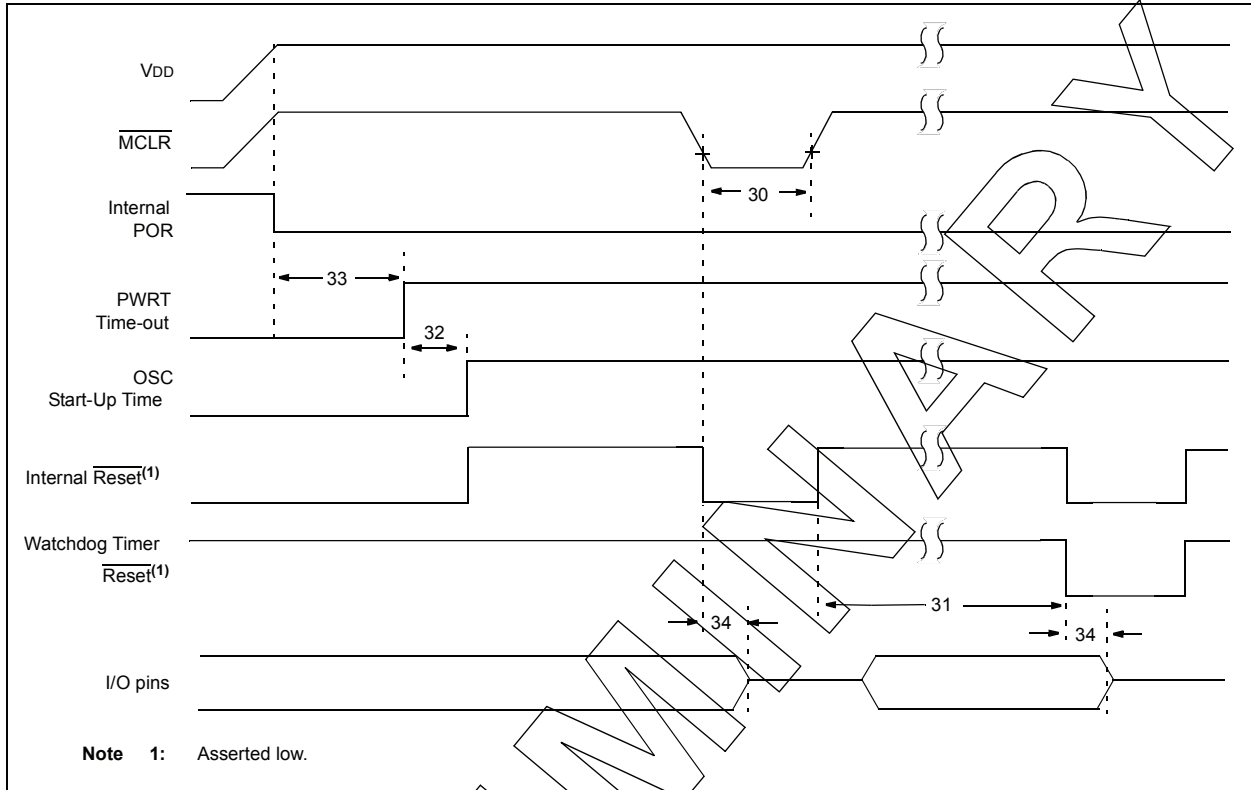
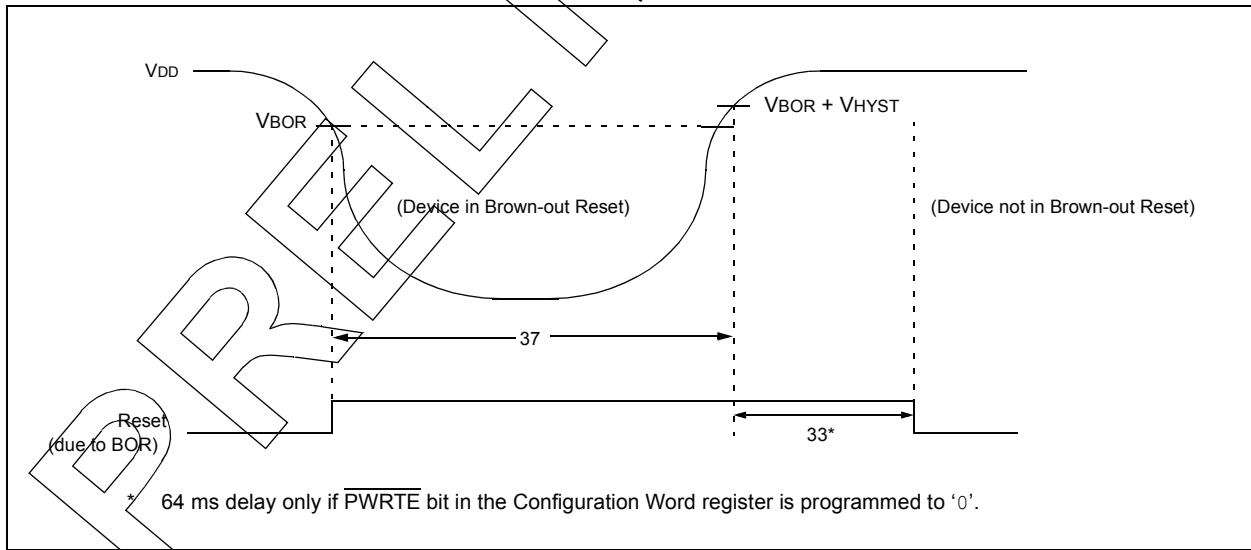


FIGURE 20-7: BROWN-OUT RESET TIMING AND CHARACTERISTICS



PIC12F752/HV752

TABLE 20-4: RESET, WATCHDOG TIMER, OSCILLATOR START-UP TIMER, POWER-UP TIMER AND BROWN-OUT RESET PARAMETERS

| Standard Operating Conditions (unless otherwise stated) | | | | | | | |
|--|-------|--|-----|------|-----|---------------|--|
| Operating Temperature $-40^{\circ}\text{C} \leq T_A \leq +125^{\circ}\text{C}$ | | | | | | | |
| Param No. | Sym | Characteristic | Min | Typ† | Max | Units | Conditions |
| 30 | TMCL | MCLR Pulse Width (low) | 2 | — | — | μs | $V_{DD} = 5\text{V}, -40^{\circ}\text{C to } +85^{\circ}\text{C}$ |
| | | | 5 | — | — | μs | $V_{DD} = 5\text{V}, -40^{\circ}\text{C to } +125^{\circ}\text{C}$ |
| 31* | TWDT | Watchdog Timer Time-out Period (No Prescaler) | 10 | 20 | TBD | ms | $V_{DD} = 5\text{V}, -40^{\circ}\text{C to } +85^{\circ}\text{C}$ |
| | | | 10 | 20 | TBD | ms | $V_{DD} = 5\text{V}, -40^{\circ}\text{C to } +125^{\circ}\text{C}$ |
| 32* | TPWRT | Power-up Timer Period | 40 | 65 | 140 | ms | |
| 33* | TIOZ | I/O High-impedance from MCLR Low or Watchdog Timer Reset | — | — | 2.0 | μs | |
| 34 | VBOR | Brown-out Reset Voltage | 2.0 | 2.15 | 2.3 | V | (NOTE 4) |
| 35* | VHYST | Brown-out Reset Hysteresis | — | 100 | — | mV | |
| 36* | TBOR | Brown-out Reset Minimum Detection Period | 100 | — | — | μs | $V_{DD} \leq V_{BOR}$ |

Legend: TBD = To Be Determined

* These parameters are characterized but not tested.

† 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: Instruction cycle period (T_{CY}) equals four times the input oscillator time base period. All specified values are based on characterization data for that particular oscillator type under standard operating conditions with the device executing code. Exceeding these specified limits may result in an unstable oscillator operation and/or higher than expected current consumption. All devices are tested to operate at "min" values with an external clock applied to the CLKIN pin. When an external clock input is used, the "max" cycle time limit is "DC" (no clock) for all devices.

2: By design.

3: Period of the slower clock.

4: To ensure these voltage tolerances, V_{DD} and V_{SS} must be capacitively decoupled as close to the device as possible. 0.1 μF and 0.01 μF values in parallel are recommended.

PRELIMINARY

FIGURE 20-8: TIMER0 AND TIMER1 EXTERNAL CLOCK TIMINGS

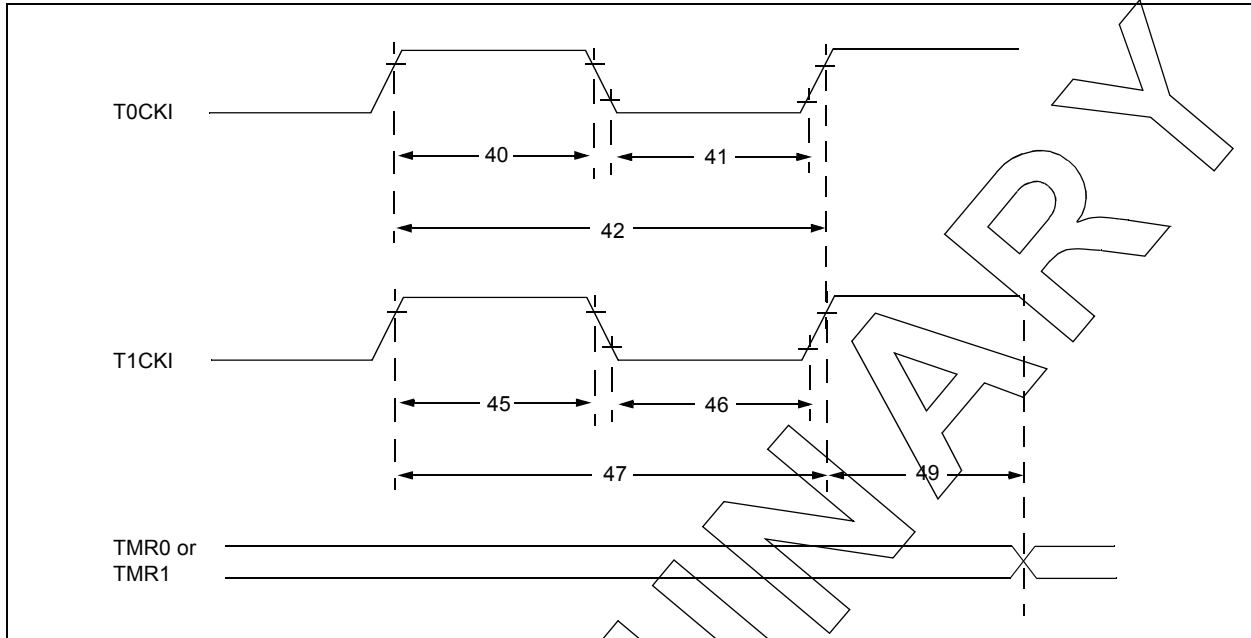


TABLE 20-5: TIMER0 AND TIMER1 EXTERNAL CLOCK REQUIREMENTS

| Standard Operating Conditions (unless otherwise stated) | | | | | | | | |
|--|-----------|---|-----------------------------|--|------|-------------|-------|-------------------------------------|
| Operating Temperature $-40^{\circ}\text{C} \leq T_A \leq +125^{\circ}\text{C}$ | | | | | | | | |
| Param No. | Sym | Characteristic | | Min | Typ† | Max | Units | Conditions |
| 40* | Tt0H | T0CKI High Pulse Width | No Prescaler | $0.5 T_{CY} + 20$ | — | — | ns | |
| | | | With Prescaler | 10 | — | — | ns | |
| 41* | Tt0L | T0CKI Low Pulse Width | No Prescaler | $0.5 T_{CY} + 20$ | — | — | ns | |
| | | | With Prescaler | 10 | — | — | ns | |
| 42* | Tt0P | T0CKI Period | | Greater of: 20 or $\frac{T_{CY} + 40}{N}$ | — | — | ns | N = prescale value (2, 4, ..., 256) |
| 45* | Tt1H | T1CKI High Time | Synchronous, No Prescaler | $0.5 T_{CY} + 20$ | — | — | ns | |
| | | | Synchronous, with Prescaler | 15 | — | — | ns | |
| | | | Asynchronous | 30 | — | — | ns | |
| 46* | Tt1L | T1CKI Low Time | Synchronous, No Prescaler | $0.5 T_{CY} + 20$ | — | — | ns | |
| | | | Synchronous, with Prescaler | 15 | — | — | ns | |
| | | | Asynchronous | 30 | — | — | ns | |
| 47* | Tt1P | T1CKI Input Period | Synchronous | Greater of: 30 or $\frac{T_{CY} + 40}{N}$ | — | — | ns | N = prescale value (1, 2, 4, 8) |
| | | | Asynchronous | 60 | — | — | ns | |
| 49* | TCKEZTMR1 | Delay from External Clock Edge to Timer Increment | | $2 T_{OSC}$ | — | $7 T_{OSC}$ | — | Timers in Sync mode |

* These parameters are characterized but not tested.

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

PIC12F752/HV752

FIGURE 20-9: PIC12F752/HV752 CAPTURE/COMPARE/PWM TIMINGS (CCP)

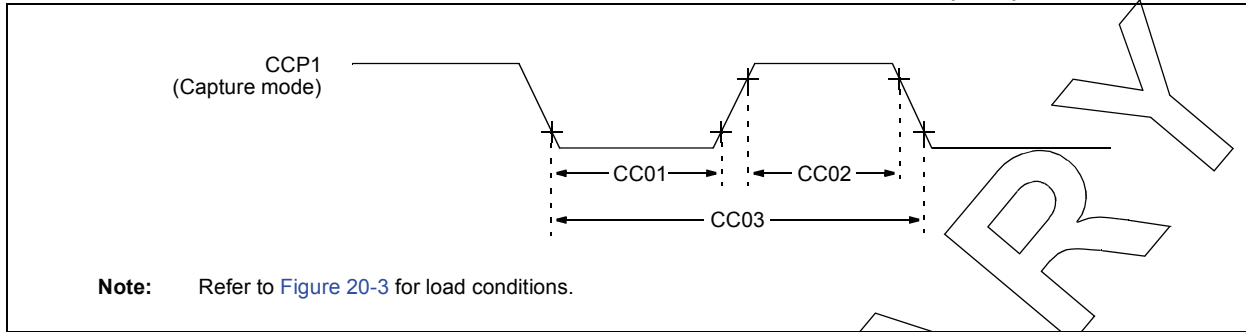


TABLE 20-6: PIC12F752/HV752 CAPTURE/COMPARE/PWM REQUIREMENTS (CCP)

| Standard Operating Conditions (unless otherwise stated) | | | | | | | | |
|--|------|----------------------|----------------|--------------------------|------|-----|-------|---------------------------------|
| Operating Temperature $-40^{\circ}\text{C} \leq T_A \leq +125^{\circ}\text{C}$ | | | | | | | | |
| Param No. | Sym | Characteristic | | Min | Typ† | Max | Units | Conditions |
| CC01* | TccL | CCP1 Input Low Time | No Prescaler | $0.5T_{CY} + 20$ | — | — | ns | |
| | | | With Prescaler | 20 | — | — | ns | |
| CC02* | TccH | CCP1 Input High Time | No Prescaler | $0.5T_{CY} + 20$ | — | — | ns | |
| | | | With Prescaler | 20 | — | — | ns | |
| CC03* | TccP | CCP1 Input Period | | $\frac{3T_{CY} + 40}{N}$ | — | — | ns | N = prescale value (1, 4 or 16) |

* These parameters are characterized but not tested.

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

TABLE 20-7: COMPARATOR SPECIFICATIONS

| Standard Operating Conditions (unless otherwise stated) | | | | | | | | | |
|--|---------|--|--|----------|-----------|----------------|---------------|----------------------|--|
| Operating Temperature $-40^{\circ}\text{C} \leq T_A \leq +125^{\circ}\text{C}$ | | | | | | | | | |
| Param No. | Sym | Characteristics | | Min | Typ† | Max | Units | Comments | |
| CM01 | VOS | Input Offset Voltage ⁽²⁾ | | — | ± 5.0 | ± 10 | mV | CxSP = 1 CxSP = 0 | |
| | | | | — | ± 7.0 | ± 20 | mV | | |
| CM02 | VCM | Input Common Mode Voltage | | 0 | — | $V_{DD} - 1.5$ | V | | |
| CM03* | CMRR | Common Mode Rejection Ratio | | +55 | — | — | dB | | |
| CM04* | TRT | Response Time ⁽¹⁾ | | CxSP = 1 | — | 55 | TBD | ns | |
| | | | | CxSP = 0 | — | 65 | TBD | ns | |
| CM05* | Tmc2coV | Comparator Mode Change to Output Valid | | — | — | 10 | μs | | |
| CM06* | VHYS | Input Hysteresis Voltage | | — | 20 | TBD | mV | | |

Legend: TBD = To Be Determined

* These parameters are characterized but not tested.

† 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: Response time is measured with one comparator input at $(V_{DD} - 1.5)/2 - 100 \text{ mV}$ to $(V_{DD} - 1.5)/2 + 20 \text{ mV}$. The other input is at $(V_{DD} - 1.5)/2$.

Note 2: Input offset voltage is measured with one comparator input at $(V_{DD} - 1.5V)/2$.

TABLE 20-8: DIGITAL-TO-ANALOG (DAC) SPECIFICATIONS

| Standard Operating Conditions (unless otherwise stated) Operating temperature $-40^{\circ}\text{C} \leq T_A \leq +125^{\circ}\text{C}$ | | | | | | | |
|---|------|-------------------------|-----|-------------|-----|---------------|----------|
| Param No. | Sym | Characteristics | Min | Typ† | Max | Units | Comments |
| DA01* | CLSB | Step Size | — | $V_{DD}/32$ | — | V | |
| DA02* | CACC | Absolute Accuracy | — | $\pm 1/2$ | — | LSB | |
| DA03* | CR | Unit Resistor Value (R) | — | TBD | — | Ω | |
| DA04* | CST | Settling Time | — | 10 | — | μs | |

Legend: TBD = To Be Determined

* These parameters are characterized but not tested.

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

TABLE 20-9: FIXED VOLTAGE REFERENCE SPECIFICATIONS

| VR Voltage Reference Specifications | | | Standard Operating Conditions (unless otherwise stated) Operating temperature $-40^{\circ}\text{C} \leq T_A \leq +125^{\circ}\text{C}$ | | | | |
|-------------------------------------|---------|--------------------|---|-----|-----|---------------|----------|
| Param No. | Symbol | Characteristics | Min | Typ | Max | Units | Comments |
| VR01* | VFVR | FVR Voltage Output | — | 1.2 | — | V | |
| VR02* | TSTABLE | FVR Turn On Time | — | 200 | — | μs | |

* These parameters are characterized but not tested.

TABLE 20-10: SHUNT REGULATOR SPECIFICATIONS (PIC12HV752 only)

| SHUNT REGULATOR CHARACTERISTICS | | | Standard Operating Conditions (unless otherwise stated) Operating temperature $-40^{\circ}\text{C} \leq T_A \leq +125^{\circ}\text{C}$ | | | | |
|---------------------------------|---------------------|-----------------------------|---|-----|-----|---------------|-------------------------------------|
| Param No. | Symbol | Characteristics | Min | Typ | Max | Units | Comments |
| SR01 | VSHUNT | Shunt Voltage | 4.75 | 5 | 5.4 | V | |
| SR02 | ISHUNT | Shunt Current | 4 | — | 50 | mA | |
| SR03* | TSETTLE | Settling Time | — | — | 150 | ns | To 1% of final value |
| SR04 | CLOAD | Load Capacitance | 0.01 | — | 10 | μF | Bypass capacitor on VDD pin |
| SR05 | ΔISNT | Regulator operating current | — | 180 | — | μA | Includes band gap reference current |

* These parameters are characterized but not tested.

PIC12F752/HV752

TABLE 20-11: PIC12F752/HV752 A/D CONVERTER (ADC) CHARACTERISTICS:

| Standard Operating Conditions (unless otherwise stated) | | | | | | | |
|--|------|--|------------|------|----------|---------------|---|
| Operating temperature $-40^{\circ}\text{C} \leq T_A \leq +125^{\circ}\text{C}$ | | | | | | | |
| Param No. | Sym | Characteristic | Min | Typ† | Max | Units | Conditions |
| AD01 | NR | Resolution | — | — | 10 bits | bit | |
| AD02 | EIL | Integral Error | — | — | ± 1 | LSb | $V_{REF} = 5.12\text{V}^{(5)}$ |
| AD03 | EDL | Differential Error | — | — | ± 1 | LSb | No missing codes to 10 bits $V_{REF} = 5.12\text{V}^{(5)}$ |
| AD04 | EOFF | Offset Error | — | +1.5 | +2.0 | LSb | $V_{REF} = 5.12\text{V}^{(5)}$ |
| AD07 | EGN | Gain Error | — | — | ± 1 | LSb | $V_{REF} = 5.12\text{V}^{(5)}$ |
| AD06 AD06A | VREF | Reference Voltage ⁽³⁾ | 2.2 2.5 | — | — VDD | V | Absolute minimum to ensure 1 LSb accuracy |
| AD07 | VAIN | Full-Scale Range | VSS | — | VREF | V | |
| AD08 | ZAIN | Recommended Impedance of Analog Voltage Source | — | — | 10 | k Ω | |
| AD09* | IREF | VREF Input Current ⁽³⁾ | 10 | — | 1000 | μA | During VAIN acquisition. Based on differential of VHOLD to VAIN. |
| | | | — | — | 50 | μA | During A/D conversion cycle. |

* These parameters are characterized but not tested.

† Data in "Typ" column is at 5.0V, 25°C unless otherwise stated. These parameters are for design guidance only and are not tested.

- Note 1:** Total Absolute Error includes integral, differential, offset and gain errors.
- Note 2:** The A/D conversion result never decreases with an increase in the input voltage and has no missing codes.
- Note 3:** ADC VREF is from external VREF or VDD pin, whichever is selected as reference input.
- Note 4:** When ADC is off, it will not consume any current other than leakage current. The power-down current specification includes any such leakage from the ADC module.
- Note 5:** VREF = 5V for PIC12HV752.

TABLE 20-12: PIC12F752/HV752 A/D CONVERSION REQUIREMENTS

| Standard Operating Conditions (unless otherwise stated) Operating temperature $-40^{\circ}\text{C} \leq T_A \leq +125^{\circ}\text{C}$ | | | | | | | |
|---|------|---|-----|--------|-----|---------------|---|
| Param No. | Sym | Characteristic | Min | Typ† | Max | Units | Conditions |
| AD130* | TAD | A/D Clock Period | 1.6 | — | 9.0 | μs | TOSC-based, $V_{\text{REF}} \geq 3.0\text{V}$ |
| | | | 3.0 | — | 9.0 | μs | TOSC-based, V_{REF} full range ⁽³⁾ |
| | | A/D Internal RC Oscillator Period | 3.0 | 6.0 | 9.0 | μs | $\text{ADCS}\langle 2:0 \rangle = 11$ At $V_{\text{DD}} = 2.5\text{V}$ |
| | | | 1.6 | 4.0 | 6.0 | μs | At $V_{\text{DD}} = 5.0\text{V}$ |
| AD131 | TCNV | Conversion Time (not including Acquisition Time) ⁽¹⁾ | — | 11 | — | TAD | Set GO/DONE bit to new data in A/D Result register |
| AD132* | TACQ | Acquisition Time | — | 11.5 | — | μs | |
| AD133* | TAMP | Amplifier Settling Time | — | — | 5 | μs | |
| AD134 | TGO | Q4 to A/D Clock Start | — | Tosc/2 | — | — | |

* These parameters are characterized but not tested.

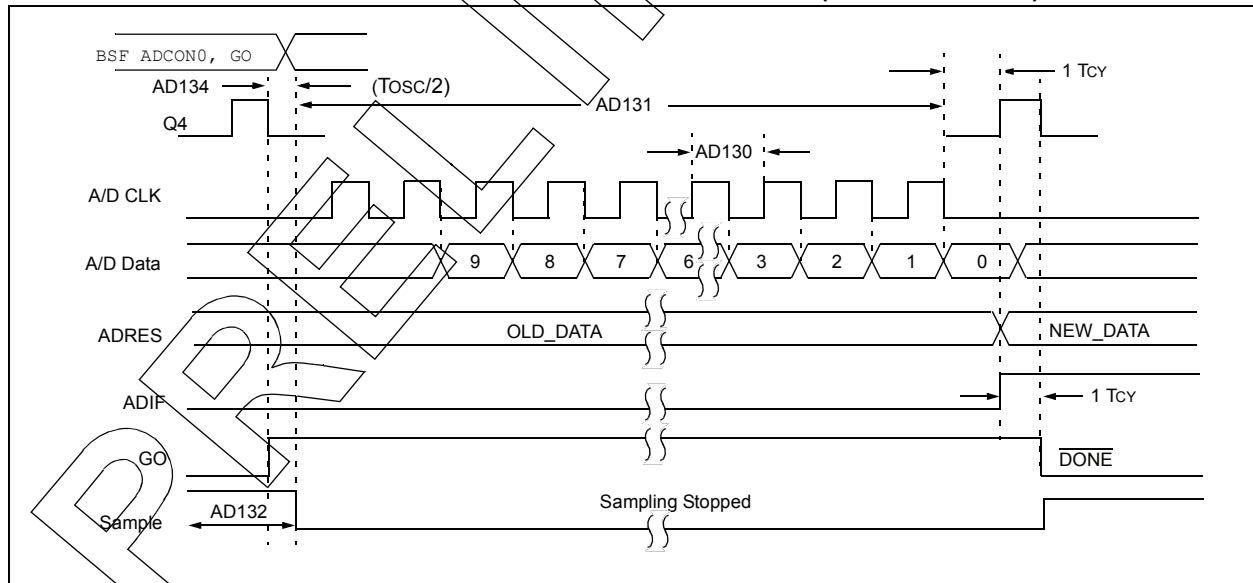
† Data in "Typ" column is at 5.0V, 25°C unless otherwise stated. These parameters are for design guidance only and are not tested.

Note 1: ADRESH and ADRESL registers may be read on the following Tcy cycle.

2: See [Section 12.4 "A/D Acquisition Requirements"](#) for minimum conditions.

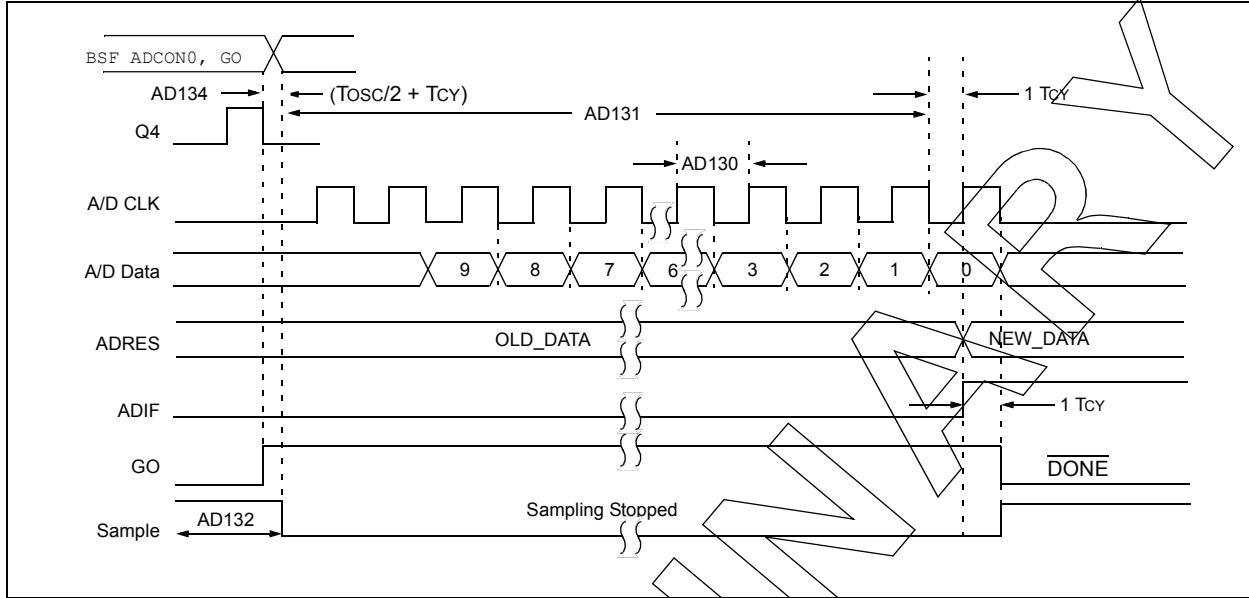
3: Full range for PIC12HV752 powered by the shunt regulator is the 5V regulated voltage.

FIGURE 20-10: PIC12F752/HV752 A/D CONVERSION TIMING (NORMAL MODE)



PIC12F752/HV752

FIGURE 20-11: PIC12F752/HV752 A/D CONVERSION TIMING (SLEEP MODE)



PRELIMINARY

21.0 DC AND AC CHARACTERISTICS GRAPHS AND CHARTS

Graphs and charts are not available at this time.

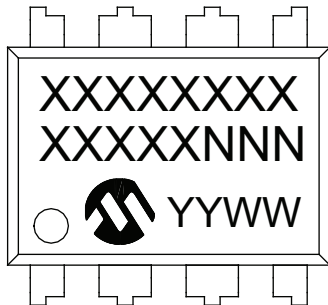
PIC12F752/HV752

NOTES:

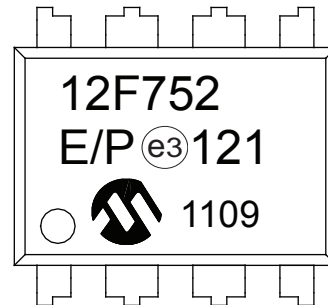
22.0 PACKAGING INFORMATION

22.1 Package Marking Information

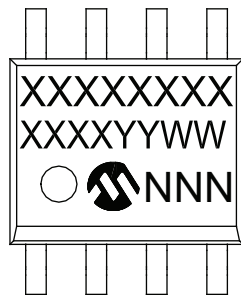
8-Lead PDIP (300 mil)



Example



8-Lead SOIC (3.90 mm)



Example



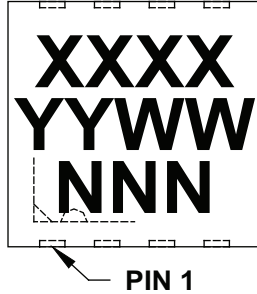
| | | |
|----------------|---|--|
| Legend: | XX...X | Customer-specific information |
| | Y | Year code (last digit of calendar year) |
| | YY | Year code (last 2 digits of calendar year) |
| | WW | Week code (week of January 1 is week '01') |
| | NNN | Alphanumeric traceability code |
| | (e3) | Pb-free JEDEC designator for Matte Tin (Sn) |
| | * | This package is Pb-free. The Pb-free JEDEC designator () (e3) can be found on the outer packaging for this package. |
| 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 PIC device marking consists of Microchip part number, year code, week code, and traceability code. For PIC 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.

PIC12F752/HV752

22.2 Package Marking Information

8-Lead DFN (3x3x0.9 mm)



Example

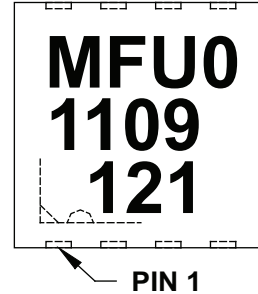





TABLE 22-1: 8-LEAD 3X3 DFN (MF) TOP MARKING

| Part Number | Marking |
|-----------------|---------|
| PIC12F752-E/MF | MFU0 |
| PIC12F752-I/MF | MFV0 |
| PIC12HV752-E/MF | MFW0 |
| PIC12HV752-I/MF | MFX0 |

| | |
|---|---|
| <p>Legend:</p> <p>XX...X</p> <p>Y</p> <p>YY</p> <p>WW</p> <p>NNN</p> <p></p> <p>* </p> | <p>Customer-specific information</p> <p>Year code (last digit of calendar year)</p> <p>Year code (last 2 digits of calendar year)</p> <p>Week code (week of January 1 is week '01')</p> <p>Alphanumeric traceability code</p> <p>Pb-free JEDEC designator for Matte Tin (Sn)</p> <p>This package is Pb-free. The Pb-free JEDEC designator ()  can be found on the outer packaging for this package.</p> |
| <p>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.</p> | |

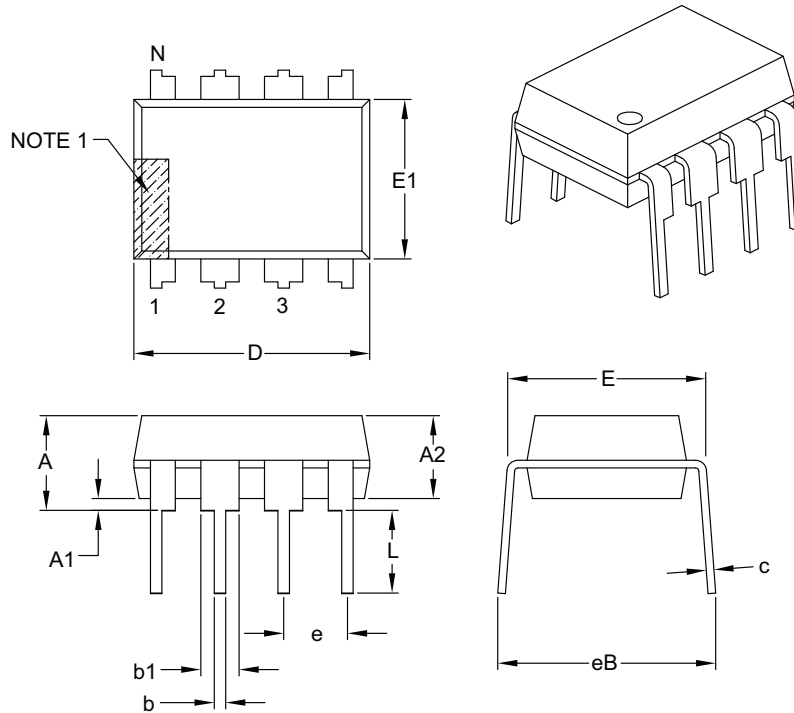
* Standard PIC device marking consists of Microchip part number, year code, week code, and traceability code. For PIC device marking beyond this, certain price adders apply. Please check with your Microchip Sales Office. For QTP devices, any special marking adders are included in QTP price.

22.3 Package Details

The following sections give the technical details of the packages.

8-Lead Plastic Dual In-Line (P) – 300 mil Body [PDIP]

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



| Dimension Limits | Units | INCHES | | |
|----------------------------|-------|----------|------|------|
| | | MIN | NOM | MAX |
| Number of Pins | N | 8 | | |
| Pitch | e | .100 BSC | | |
| Top to Seating Plane | A | – | – | .210 |
| Molded Package Thickness | A2 | .115 | .130 | .195 |
| Base to Seating Plane | A1 | .015 | – | – |
| Shoulder to Shoulder Width | E | .290 | .310 | .325 |
| Molded Package Width | E1 | .240 | .250 | .280 |
| Overall Length | D | .348 | .365 | .400 |
| Tip to Seating Plane | L | .115 | .130 | .150 |
| Lead Thickness | c | .008 | .010 | .015 |
| Upper Lead Width | b1 | .040 | .060 | .070 |
| Lower Lead Width | b | .014 | .018 | .022 |
| Overall Row Spacing § | eB | – | – | .430 |

Notes:

- Pin 1 visual index feature may vary, but must be located with the hatched area.
- § Significant Characteristic.
- Dimensions D and E1 do not include mold flash or protrusions. Mold flash or protrusions shall not exceed .010" per side.
- Dimensioning and tolerancing per ASME Y14.5M.

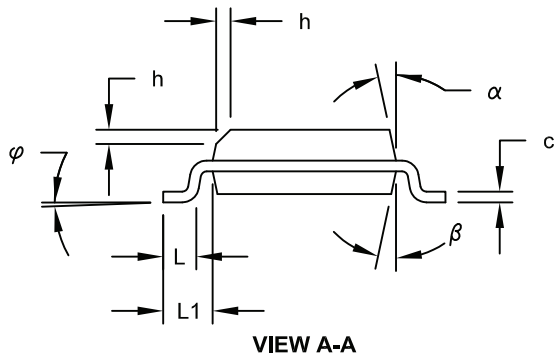
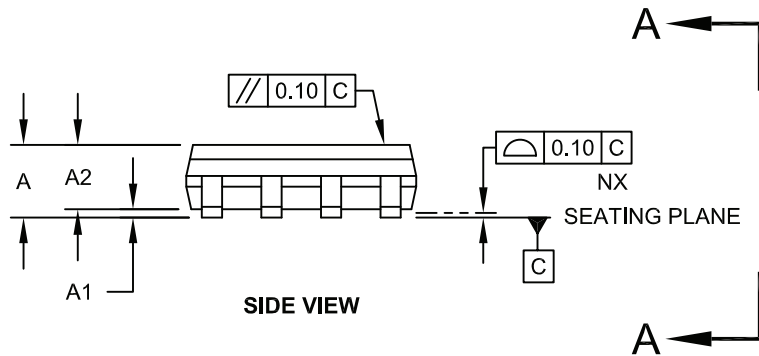
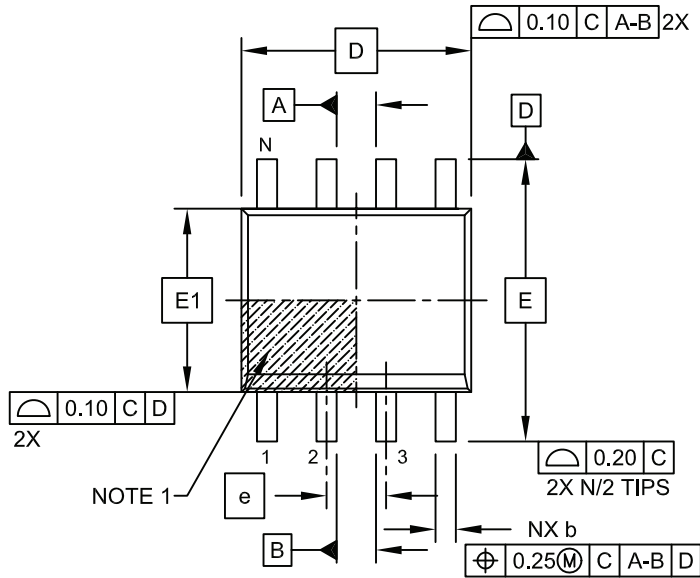
BSC: Basic Dimension. Theoretically exact value shown without tolerances.

Microchip Technology Drawing C04-018B

PIC12F752/HV752

8-Lead Plastic Small Outline (SN) - Narrow, 3.90 mm Body [SOIC]

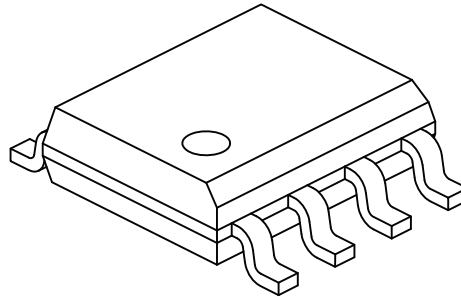
Note: For the most current package drawings, please see the Microchip Packaging Specification located at <http://www.microchip.com/packaging>



Microchip Technology Drawing No. C04-057C Sheet 1 of 2

8-Lead Plastic Small Outline (SN) - Narrow, 3.90 mm Body [SOIC]

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



| Dimension Limits | Units | MILLIMETERS | | |
|--------------------------|-----------|-------------|-----|------|
| | | MIN | NOM | MAX |
| Number of Pins | N | 8 | | |
| Pitch | e | 1.27 BSC | | |
| Overall Height | A | - | - | 1.75 |
| Molded Package Thickness | A2 | 1.25 | - | - |
| Standoff § | A1 | 0.10 | - | 0.25 |
| Overall Width | E | 6.00 BSC | | |
| Molded Package Width | E1 | 3.90 BSC | | |
| Overall Length | D | 4.90 BSC | | |
| Chamfer (Optional) | h | 0.25 | - | 0.50 |
| Foot Length | L | 0.40 | - | 1.27 |
| Footprint | L1 | 1.04 REF | | |
| Foot Angle | φ | 0° | - | 8° |
| Lead Thickness | c | 0.17 | - | 0.25 |
| Lead Width | b | 0.31 | - | 0.51 |
| Mold Draft Angle Top | α | 5° | - | 15° |
| Mold Draft Angle Bottom | β | 5° | - | 15° |

Notes:

1. Pin 1 visual index feature may vary, but must be located within the hatched area.
2. § Significant Characteristic
3. Dimensions D and E1 do not include mold flash or protrusions. Mold flash or protrusions shall not exceed 0.15mm per side.
4. Dimensioning and tolerancing per ASME Y14.5M

BSC: Basic Dimension, Theoretically exact value shown without tolerances.

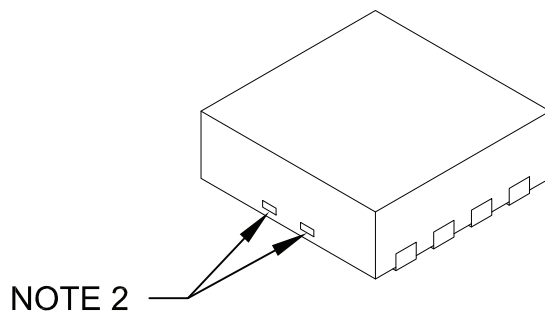
REF: Reference Dimension, usually without tolerance, for information purposes only.

Microchip Technology Drawing No. C04-057C Sheet 2 of 2

PIC12F752/HV752

8-Lead Plastic Dual Flat, No Lead Package (MF) - 3x3x0.9mm Body [DFN]

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



| Dimension Limits | Units | MILLIMETERS | | |
|------------------------|-------|-------------|------|------|
| | | MIN | NOM | MAX |
| Number of Pins | N | 8 | | |
| Pitch | e | 0.65 BSC | | |
| Overall Height | A | 0.80 | 0.90 | 1.00 |
| Standoff | A1 | 0.00 | 0.02 | 0.05 |
| Contact Thickness | A3 | 0.20 REF | | |
| Overall Length | D | 3.00 BSC | | |
| Exposed Pad Width | E2 | 1.34 | - | 1.60 |
| Overall Width | E | 3.00 BSC | | |
| Exposed Pad Length | D2 | 1.60 | - | 2.40 |
| Contact Width | b | 0.25 | 0.30 | 0.35 |
| Contact Length | L | 0.20 | 0.30 | 0.55 |
| Contact-to-Exposed Pad | K | 0.20 | - | - |

Notes:

1. Pin 1 visual index feature may vary, but must be located within the hatched area.
2. Package may have one or more exposed tie bars at ends.
3. Package is saw singulated
4. Dimensioning and tolerancing per ASME Y14.5M

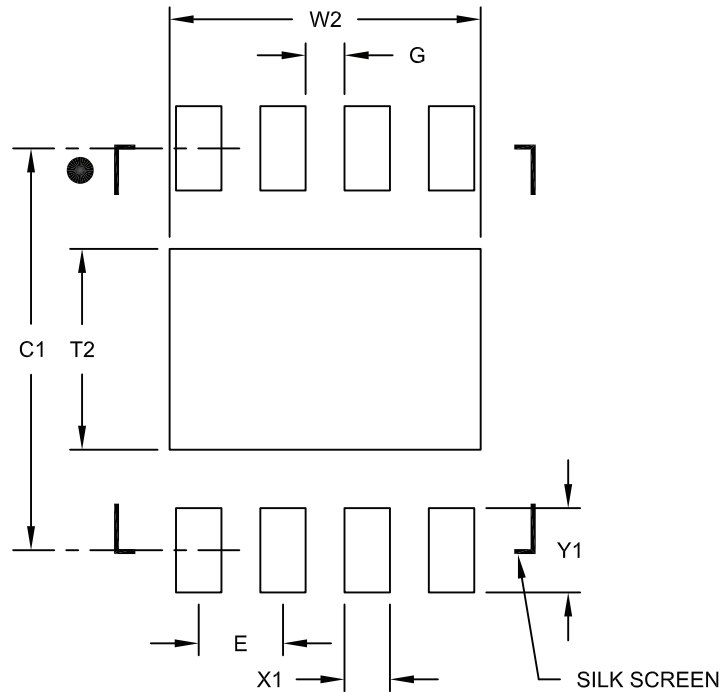
BSC: Basic Dimension. Theoretically exact value shown without tolerances.

REF: Reference Dimension, usually without tolerance, for information purposes only.

Microchip Technology Drawing No. C04-062C Sheet 2 of 2

8-Lead Plastic Dual Flat, No Lead Package (MF) – 3x3x0.9 mm Body [DFN]

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



RECOMMENDED LAND PATTERN

| Dimension Limits | Units | MILLIMETERS | | |
|----------------------------|-------|-------------|------|------|
| | | MIN | NOM | MAX |
| Contact Pitch | E | 0.65 BSC | | |
| Optional Center Pad Width | W2 | | | 2.40 |
| Optional Center Pad Length | T2 | | | 1.55 |
| Contact Pad Spacing | C1 | | 3.10 | |
| Contact Pad Width (X8) | X1 | | | 0.35 |
| Contact Pad Length (X8) | Y1 | | | 0.65 |
| Distance Between Pads | G | 0.30 | | |

Notes:

1. Dimensioning and tolerancing per ASME Y14.5M

BSC: Basic Dimension. Theoretically exact value shown without tolerances.

Microchip Technology Drawing No. C04-2062A

PIC12F752/HV752

NOTES:

APPENDIX A: DATA SHEET REVISION HISTORY

Revision A

Original release (4/2011).

Revision B

Redefined operation of the COG module; Added slew rate control to the COG module; Added zero latency filter to the comparator; Updated Electrical Specifications.

APPENDIX B: MIGRATING FROM PIC12HV615

This compares the features of the PIC12HV615 to the PIC12HV752 family of devices.

B.1 PIC12HV615 to PIC12HV752

TABLE B-1: FEATURE COMPARISON

| Feature | PIC12HV615 | PIC12HV752 |
|---|---------------|-------------------------|
| Max Operating Speed | 20 MHz | 20 MHz |
| Max Program Memory (Words) | 1024 | 1024 |
| Flash Self Read/ Self Write | No | Yes |
| SRAM (bytes) | 64 | 64 |
| Oscillator modes | 8 | 2 |
| INTOSC Frequencies | 4/8 MHz | 1/4/8 MHz and 31 kHz |
| Brown-out Reset (BOR) | Y | Y |
| Internal Pull-ups | GP0/1/2/3/4/5 | RA0/1/2/3/4/5 |
| Interrupt-on-change | GP0/1/2/3/4/5 | RA0/1/2/3/4/5 |
| Analog-to-Digital Converter (ADC) Channels | 4 | 4 |
| A/D Resolution | 10-bit | 10-bit |
| Timers (8/16-bit) | 2/1 | 3/1 |
| Comparator | 1 | 2 High Speed |
| ECCP/CCP | 1/0 | 0/1 |
| Complementary Output Generator (COG) | No | Yes |
| Digital-to-Analog Converter (DAC) 5-bit Dual Range | No | Yes |
| Fixed Voltage Reference (FVR) | No | Yes |
| Internal Shunt Regulator | Yes | Yes |

Note: This device has been designed to perform to the parameters of its data sheet. It has been tested to an electrical specification designed to determine its conformance with these parameters. Due to process differences in the manufacture of this device, this device may have different performance characteristics than its earlier version. These differences may cause this device to perform differently in your application than the earlier version of this device.

PIC12F752/HV752

NOTES:

INDEX

A

| | |
|---|----------|
| A/D | |
| Specifications | 188, 189 |
| Absolute Maximum Ratings | 167 |
| AC Characteristics | |
| Industrial and Extended | 180 |
| Load Conditions | 179 |
| ADC | 101 |
| Acquisition Requirements | 109 |
| Associated registers | 111 |
| Block Diagram | 101 |
| Calculating Acquisition Time | 109 |
| Channel Selection | 102 |
| Configuration | 102 |
| Configuring Interrupt | 105 |
| Conversion Clock | 102 |
| Conversion Procedure | 105 |
| Internal Sampling Switch (R _{ss}) Impedance | 109 |
| Interrupts | 103 |
| Operation | 104 |
| Operation During Sleep | 104 |
| Port Configuration | 102 |
| Reference Voltage (V _{REF}) | 102 |
| Result Formatting | 104 |
| Source Impedance | 109 |
| Special Event Trigger | 104 |
| Starting an A/D Conversion | 104 |
| ADCON0 Register | 106 |
| ADCON1 Register | 107 |
| ADRESH Register (ADFM = 0) | 108 |
| ADRESH Register (ADFM = 1) | 108 |
| ADRESL Register (ADFM = 0) | 108 |
| ADRESL Register (ADFM = 1) | 108 |
| Alternate Pin Function | 46 |
| Analog-to-Digital Converter. <i>See</i> ADC | |
| ANSELA Register | 50 |
| APFCON Register | 46 |
| Assembler | |
| MPASM Assembler | 164 |

B

| | |
|---|----------------|
| Block Diagrams | |
| (CCP) Capture Mode Operation | 77 |
| ADC | 101 |
| ADC Transfer Function | 110 |
| Analog Input Model | 110, 126 |
| CCP PWM | 81 |
| Clock Source | 39 |
| Compare | 79 |
| Digital-to-Analog Converter (DAC) | 115 |
| Generic I/O Port | 45 |
| HLTMR1 | 73 |
| In-Circuit Serial Programming Connections | 160 |
| Interrupt Logic | 153 |
| MCLR Circuit | 145 |
| On-Chip Reset Circuit | 143 |
| PIC12F752/HV752 | 9 |
| Timer1 | 59, 64, 65, 66 |
| Timer2 | 71 |
| TMR0/WDT Prescaler | 55 |
| Voltage Reference | 113 |
| Voltage Reference Output Buffer Example | 117 |
| Watchdog Timer | 156 |

| | |
|----------------------------|-----|
| Brown-out Reset (BOR) | 146 |
| Associated Registers | 146 |
| Specifications | 184 |
| Timing and Characteristics | 183 |

C

| | |
|--|--------|
| C Compilers | |
| MPLAB C18 | 164 |
| Calibration Bits | 143 |
| Capture Module. <i>See</i> Capture/Compare/PWM (CCP) | |
| Capture/Compare/PWM | 77 |
| Capture/Compare/PWM (CCP) | 77 |
| Associated Registers w/ Capture | 78 |
| Associated Registers w/ Compare | 80 |
| Capture Mode | 77 |
| CCP1 Pin Configuration | 77 |
| Compare Mode | 79 |
| CCP1 Pin Configuration | 79 |
| Software Interrupt Mode | 77, 79 |
| Special Event Trigger | 79 |
| Timer1 Mode Resource | 77, 79 |
| Prescaler | 77 |
| PWM Mode | |
| Duty Cycle | 82 |
| PWM Operation | 81 |
| PWM Overview | 81 |
| PWM Period | 82 |
| PWM Setup | 82 |
| Specifications | 186 |
| CCP. <i>See</i> Capture/Compare/PWM | |
| CCP1CON (CCP1) Register | 83 |
| Clock Sources | |
| External Modes | |
| EC | 40 |
| Internal Modes | 40 |
| CMOUT Register | 128 |
| CMxCON0 Register | 127 |
| CMxCON1 Register | 128 |
| Code Examples | |
| A/D Conversion | 105 |
| Assigning Prescaler to Timer0 | 56 |
| Assigning Prescaler to WDT | 56 |
| Changing Between Capture Prescalers | 77 |
| Indirect Addressing | 26 |
| Initializing PORTA | 45 |
| Saving Status and W Registers in RAM | 155 |
| Writing to Flash Program Memory | 36 |
| Code Protection | 159 |
| COGxASD Register | 97 |
| COGxBLK Register | 98 |
| COGxCON0 Register | 95 |
| COGxCON1 Register | 96 |
| COGxDB Register | 98 |
| COGxPH Register | 98 |
| Comparator | 121 |
| Associated Registers | 129 |
| Operation | 121 |
| Comparator Module | 121 |
| Cx Output State Versus Input Conditions | 123 |
| Comparators | |
| C2OUT as T1 Gate | 61 |
| Specifications | 186 |
| Compare Module. <i>See</i> Capture/Compare/PWM (CCP) | |
| CONFIG1 Register | 142 |

PIC12F752/HV752

| | | | |
|---|-----|---|----------|
| Configuration Bits | 141 | CLRW | 134 |
| CPU Features | 141 | CLRWDT | 134 |
| Customer Change Notification Service | 207 | COMF | 134 |
| Customer Notification Service | 207 | DECf | 134 |
| Customer Support | 207 | DECFSZ | 135 |
| D | | GOTO | 135 |
| DACCON0 (Digital-to-Analog Converter Control 0) Register | 118 | INCF | 135 |
| DACCON1 (Digital-to-Analog Converter Control 1) Register | 118 | INCFSZ | 135 |
| Data Memory | 11 | IORLW | 135 |
| DC Characteristics | | IORWF | 135 |
| Extended and Industrial | 176 | MOVLW | 136 |
| Industrial and Extended | 169 | MOVWF | 136 |
| Development Support | 163 | NOP | 136 |
| Device Overview | 9 | RETFIE | 137 |
| Digital-to-Analog (DAC) | | RETLW | 137 |
| Specifications | 187 | RETURN | 137 |
| Digital-to-Analog Converter (DAC) | 115 | RLF | 138 |
| Associated Registers | 119 | RRF | 138 |
| Effects of a Reset | 117 | SLEEP | 138 |
| E | | SUBLW | 138 |
| Electrical Specifications | 167 | SUBWF | 139 |
| Errata | 7 | SWAPF | 139 |
| F | | XORLW | 139 |
| Firmware Instructions | 131 | XORWF | 139 |
| Fixed Voltage Reference (FVR) | | Summary Table | 132 |
| Associated Registers | 114 | INTCON Register | 20 |
| Specifications | 187 | Internal Oscillator Block | |
| Flash Program Memory | | INTOSC | |
| Associated Registers | 37 | Specifications | 181, 182 |
| Configuration Word w/ Flash Program Memory | 37 | Internal Sampling Switch (R _{ss}) Impedance | 109 |
| Flash Program Memory Self Read/Self Write Control | 29 | Internet Address | 207 |
| FVRCON (Fixed Voltage Reference Control) Register | 114 | Interrupts | 152 |
| G | | ADC | 105 |
| General Purpose Register File | 12 | Associated Registers | 154 |
| H | | Context Saving | 155 |
| Hardware Limit Timer (HLT) | 73 | Interrupt-on-Change | 49 |
| High Temperature Operation | 190 | PORTA Interrupt-on-Change | 153 |
| HLT | | RA2/INT | 152 |
| Associated registers | 76 | Timer0 | 153 |
| HLT1CON0 (HLT1 Control 0) Register | 75 | TMR1 | 63 |
| HLT1CON1 (HLT1 Control 1) Register | 76 | INTOSC Specifications | 181, 182 |
| I | | IOCAF Register | 52 |
| ID Locations | 159 | IOCAN Register | 52 |
| In-Circuit Serial Programming (ICSP) | 160 | IOCAP Register | 52 |
| Indirect Addressing, INDF and FSR registers | 26 | L | |
| Instruction Format | 131 | LATA Register | 48 |
| Instruction Set | 131 | Load Conditions | 179 |
| ADDLW | 133 | M | |
| ADDWF | 133 | MCLR | 145 |
| ANDLW | 133 | Internal | 145 |
| ANDWF | 133 | Memory Organization | 11, 85 |
| MOVf | 136 | Data | 11 |
| BCF | 133 | Program | 11 |
| BSF | 133 | Microchip Internet Web Site | 207 |
| BTFSC | 133 | Migrating from other PIC Devices | 201 |
| BTFSS | 134 | MPLAB ASM30 Assembler, Linker, Librarian | 164 |
| CALL | 134 | MPLAB Integrated Development Environment Software | 163 |
| CLRF | 134 | MPLAB PM3 Device Programmer | 166 |
| | | MPLAB REAL ICE In-Circuit Emulator System | 165 |
| | | MPLINK Object Linker/MPLIB Object Librarian | 164 |

O

| | |
|--------------------------------------|------------|
| OPCODE Field Descriptions | 131 |
| Operation During Code Protect | 34 |
| Operation During Write Protect | 34 |
| Operational Amplifier (OPA) Module | |
| AC Specifications | 187 |
| OPTION Register | 19 |
| OPTION_REG Register | 57 |
| Oscillator | |
| Associated registers | 43, 69, 99 |
| Configuration Word | 43 |
| Oscillator Module | 29, 39 |
| Oscillator Parameters | 181 |
| Oscillator Specifications | 180 |
| Oscillator Start-up Timer (OST) | |
| Specifications | 184 |
| OSCTUNE Register | 43 |

P

| | |
|--|----------|
| Packaging | 193 |
| Marking | 193, 194 |
| PDIP Details | 194 |
| PCL and PCLATH | 26 |
| Stack | 26 |
| PCON Register | 25, 147 |
| PIE1 Register | 21 |
| PIE2 Register | 22 |
| Pinout Descriptions | |
| PIC12F752/HV752 | 10 |
| PIR1 Register | 23 |
| PIR2 Register | 24 |
| PMADRH and PMADRL Registers | 29 |
| PMCON1 and PMCON2 Registers | 29 |
| PMCON1 Register | 31 |
| PMDATH Register | 30 |
| PMDATL Register | 30 |
| PMDRH Register | 30 |
| PORTA | |
| Additional Pin Functions | 49 |
| ANSEL Register | 49 |
| Interrupt-on-Change | 49 |
| Slew Rate Control | 49 |
| Weak Pull-Ups | 49 |
| Associated Registers | 53 |
| RA2 | 53 |
| RA3 | 53 |
| RA4 | 53 |
| Specifications | 182 |
| PORTA Register | 48 |
| Power-Down Mode (Sleep) | 158 |
| Power-on Reset (POR) | 144 |
| Power-up Timer (PWRT) | 145 |
| Specifications | 184 |
| Precision Internal Oscillator Parameters | 182 |
| Prescaler | |
| Shared WDT/Timer0 | 56 |
| Switching Prescaler Assignment | 56 |
| Program Memory | 11 |
| Map and Stack | 11 |
| Programming, Device Instructions | 131 |
| Protection Against Spurious Write | 34 |
| PWM (CCP Module) | |
| PWM Steering | 83 |
| PWM Steering | 83 |

R

| | |
|---|---------|
| Reader Response | 208 |
| Reading the Flash Program Memory | 32 |
| Read-Modify-Write Operations | 131 |
| Registers | |
| ADCON0 (ADC Control 0) | 106 |
| ADCON1 (ADC Control 1) | 107 |
| ADRESH (ADC Result High) with ADFM = 0 | 108 |
| ADRESH (ADC Result High) with ADFM = 1 | 108 |
| ADRESL (ADC Result Low) with ADFM = 0 | 108 |
| ADRESL (ADC Result Low) with ADFM = 1 | 108 |
| ANSELA (PORTA Analog Select) | 50 |
| APFCON (Alternate Pin Function Control) | 46 |
| CCP1CON (CCP1 Control) | 83 |
| CMOUT (Comparator Output) | 128 |
| CMxCON0 (Cx Control) | 127 |
| CMxCON1 (Cx Control 1) | 128 |
| COGxASD (COG Auto-Shutdown Control) | 97 |
| COGxBLK (COG Blanking Deadband Counter) | 98 |
| COGxCON0 (COG Control 0) | 95 |
| COGxCON1 (COG Control 1) | 96 |
| COGxDB (COG Deadband Counter) | 98 |
| COGxPH (COG Blanking Phase Counter) | 98 |
| Configuration Word 1 | 142 |
| DACCON0 | 118 |
| DACCON1 | 118 |
| Data Memory Map | 13 |
| FVRCON | 114 |
| HLT1CON0 | 75 |
| HLT1CON1 | 76 |
| INTCON (Interrupt Control) | 20 |
| IOCAF (Interrupt-on-Change Flag) | 52 |
| IOCAN (Interrupt-on-Change Negative Edge) | 52 |
| IOCAP (Interrupt-on-Change Positive Edge) | 52 |
| LATA (Data Latch PORTA) | 48 |
| OPTION_REG (OPTION) | 19, 57 |
| OSCTUNE (Oscillator Tuning) | 43 |
| PCON (Power Control) | 25, 147 |
| PIE1 (Peripheral Interrupt Enable 1) | 21 |
| PIE2 (Peripheral Interrupt Enable 2) | 22 |
| PIR1 (Peripheral Interrupt Register 1) | 23 |
| PIR2 (Peripheral Interrupt Register 2) | 24 |
| PMCON1 (Program Memory Control 1) | 31 |
| PMDATH (Program Memory Data) | 30 |
| PMDATL (Program Memory Data) | 30 |
| PMDRH (Program Memory Address) | 30 |
| PORTA | 48 |
| Reset Values | 149 |
| Reset Values (special registers) | 151 |
| Special Function Registers | 12 |
| STATUS | 18 |
| T1CON (Timer1 Control) | 67 |
| T1GCON (Timer1 Gate Control) | 68 |
| T2CON | 72 |
| TRISA (Tri-State PORTA) | 48 |
| WPUA (Weak Pull-up PORTA) | 50 |
| Reset | 143 |
| Revision History | 201 |
| S | |
| Shunt Regulator | 161 |
| Sleep | |
| Power-Down Mode | 158 |
| Wake-up | 158 |
| Wake-up using Interrupts | 158 |

PIC12F752/HV752

| | |
|--------------------------------------|-----|
| Slew Rate Control | 49 |
| Software Simulator (MPLAB SIM) | 165 |
| Special Event Trigger | 104 |
| Special Function Registers | 12 |
| STATUS Register | 18 |

T

| | |
|--|--------|
| T1CON Register | 67 |
| T1GCON Register | 68 |
| T2CON Register | 72 |
| Thermal Considerations | 178 |
| Time-out Sequence | 147 |
| Timer0 | 55 |
| Associated Registers | 57 |
| External Clock | 56 |
| Interrupt | 57 |
| Operation | 55, 60 |
| Specifications | 185 |
| T0CKI | 56 |
| Timer1 | |
| Associated registers | 69 |
| Asynchronous Counter Mode | 61 |
| Reading and Writing | 61 |
| Interrupt | 63 |
| Modes of Operation | 60 |
| Operation During Sleep | 63 |
| Prescaler | 61 |
| Specifications | 185 |
| Timer1 Gate | |
| Selecting Source | 61 |
| TMR1H Register | 59 |
| TMR1L Register | 59 |
| Timer2 | |
| Associated registers | 72, 76 |
| Timers | |
| HLT | |
| HLT1CON0 | 75 |
| HLT1CON1 | 76 |
| Timer1 | |
| T1CON | 67 |
| T1GCON | 68 |
| Timer2 | |
| T2CON | 72 |
| Timing Diagrams | |
| A/D Conversion | 189 |
| A/D Conversion (Sleep Mode) | 190 |
| Brown-out Reset (BOR) | 183 |
| Brown-out Reset Situations | 146 |
| Capture/Compare/PWM (CCP) | 186 |
| CLKOUT and I/O | 182 |
| Clock Timing | 180 |
| Comparator Output | 121 |
| INT Pin Interrupt | 154 |
| Time-out Sequence | |
| Case 1 | 147 |
| Case 2 | 147 |
| Case 3 | 148 |
| Timer0 and Timer1 External Clock | 185 |
| Timer1 Incrementing Edge | 63 |
| Wake-up from Interrupt | 159 |
| Timing Parameter Symbolology | 179 |
| TRISA Register | 48 |

V

VREF. *SEE* ADC Reference Voltage

W

| | |
|--|-----|
| Wake-up Using Interrupts | 158 |
| Watchdog Timer (WDT) | 155 |
| Associated registers | 157 |
| Configuration Word w/ Watchdog Timer | 157 |
| Specifications | 184 |
| WPUA Register | 50 |
| Writing the Flash Program Memory | 34 |
| WWW Address | 207 |
| WWW, On-Line Support | 7 |

THE MICROCHIP WEB SITE

Microchip provides online support via our WWW site at www.microchip.com. This web site is used as a means to make files and information easily available to customers. Accessible by using your favorite Internet browser, the web site contains the following information:

- **Product Support** – Data sheets and errata, application notes and sample programs, design resources, user's guides and hardware support documents, latest software releases and archived software
- **General Technical Support** – Frequently Asked Questions (FAQ), technical support requests, online discussion groups, Microchip consultant program member listing
- **Business of Microchip** – Product selector and ordering guides, latest Microchip press releases, listing of seminars and events, listings of Microchip sales offices, distributors and factory representatives

CUSTOMER CHANGE NOTIFICATION SERVICE

Microchip's customer notification service helps keep customers current on Microchip products. Subscribers will receive e-mail notification whenever there are changes, updates, revisions or errata related to a specified product family or development tool of interest.

To register, access the Microchip web site at www.microchip.com. Under "Support", click on "Customer Change Notification" and follow the registration instructions.

CUSTOMER SUPPORT

Users of Microchip products can receive assistance through several channels:

- Distributor or Representative
- Local Sales Office
- Field Application Engineer (FAE)
- Technical Support
- Development Systems Information Line

Customers should contact their distributor, representative or field application engineer (FAE) for support. Local sales offices are also available to help customers. A listing of sales offices and locations is included in the back of this document.

Technical support is available through the web site at: <http://microchip.com/support>

PIC12F752/HV752

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 document.

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: PIC12F752/HV752

Literature Number: DS41576B

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 document easy to follow? If not, why?

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

5. What deletions from the document 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?

PRODUCT IDENTIFICATION SYSTEM

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

| <u>PART NO.</u> | <u>[X]⁽¹⁾</u> | - | <u>X</u> | <u>/XX</u> | <u>XXX</u> |
|---|--|---|-------------------|------------|------------|
| Device | Tape and Reel Option | | Temperature Range | Package | Pattern |
| Device: | PIC12F752 PIC12HV752 | | | | |
| Tape and Reel Option: | Blank = Standard packaging (tube or tray) T = Tape and Reel ⁽¹⁾ | | | | |
| Temperature Range: | I = -40°C to +85°C (Industrial) E = -40°C to +125°C (Extended) | | | | |
| Package: | P = Plastic DIP (PDIP) SN = 8-lead Small Outline (150 mil) (SOIC) MF = 8-lead Plastic Dual Flat, No Lead (3x3) (DFN) | | | | |
| Pattern: | QTP, SQTP, Code or Special Requirements (blank otherwise) | | | | |
| Examples: | | | | | |
| a) PIC12F752T - I/MF 301 Tape and Reel, Industrial temperature, DFN 3x3 package, QTP pattern #301 | | | | | |
| b) PIC12F752 - E/P Extended temperature PDIP package | | | | | |
| c) PIC12F752 - E/SN Extended temperature, SOIC package | | | | | |
| d) PIC12HV752 - E/MF Extended temperature, DFN 3x3 package | | | | | |
| Note 1: Tape and Reel identifier only appears in the catalog part number description. This identifier is used for ordering purposes and is not printed on the device package. Check with your Microchip Sales Office for package availability with the Tape and Reel option. | | | | | |
| Note 1: T = in tape and reel for SOIC and DFN packages only. | | | | | |



MICROCHIP

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:
<http://www.microchip.com/support>
Web Address:
www.microchip.com

Atlanta
Duluth, GA
Tel: 678-957-9614
Fax: 678-957-1455

Boston
Westborough, MA
Tel: 774-760-0087
Fax: 774-760-0088

Chicago
Itasca, IL
Tel: 630-285-0071
Fax: 630-285-0075

Cleveland
Independence, OH
Tel: 216-447-0464
Fax: 216-447-0643

Dallas
Addison, TX
Tel: 972-818-7423
Fax: 972-818-2924

Detroit
Farmington Hills, MI
Tel: 248-538-2250
Fax: 248-538-2260

Indianapolis
Noblesville, IN
Tel: 317-773-8323
Fax: 317-773-5453

Los Angeles
Mission Viejo, CA
Tel: 949-462-9523
Fax: 949-462-9608

Santa Clara
Santa Clara, CA
Tel: 408-961-6444
Fax: 408-961-6445

Toronto
Mississauga, Ontario,
Canada
Tel: 905-673-0699
Fax: 905-673-6509

ASIA/PACIFIC

Asia Pacific Office
Suites 3707-14, 37th Floor
Tower 6, The Gateway
Harbour City, Kowloon
Hong Kong
Tel: 852-2401-1200
Fax: 852-2401-3431

Australia - Sydney
Tel: 61-2-9868-6733
Fax: 61-2-9868-6755

China - Beijing
Tel: 86-10-8569-7000
Fax: 86-10-8528-2104

China - Chengdu
Tel: 86-28-8665-5511
Fax: 86-28-8665-7889

China - Chongqing
Tel: 86-23-8980-9588
Fax: 86-23-8980-9500

China - Hangzhou
Tel: 86-571-2819-3187
Fax: 86-571-2819-3189

China - Hong Kong SAR
Tel: 852-2401-1200
Fax: 852-2401-3431

China - Nanjing
Tel: 86-25-8473-2460
Fax: 86-25-8473-2470

China - Qingdao
Tel: 86-532-8502-7355
Fax: 86-532-8502-7205

China - Shanghai
Tel: 86-21-5407-5533
Fax: 86-21-5407-5066

China - Shenyang
Tel: 86-24-2334-2829
Fax: 86-24-2334-2393

China - Shenzhen
Tel: 86-755-8203-2660
Fax: 86-755-8203-1760

China - Wuhan
Tel: 86-27-5980-5300
Fax: 86-27-5980-5118

China - Xian
Tel: 86-29-8833-7252
Fax: 86-29-8833-7256

China - Xiamen
Tel: 86-592-2388138
Fax: 86-592-2388130

China - Zhuhai
Tel: 86-756-3210040
Fax: 86-756-3210049

ASIA/PACIFIC

India - Bangalore
Tel: 91-80-3090-4444
Fax: 91-80-3090-4123

India - New Delhi
Tel: 91-11-4160-8631
Fax: 91-11-4160-8632

India - Pune
Tel: 91-20-2566-1512
Fax: 91-20-2566-1513

Japan - Yokohama
Tel: 81-45-471- 6166
Fax: 81-45-471-6122

Korea - Daegu
Tel: 82-53-744-4301
Fax: 82-53-744-4302

Korea - Seoul
Tel: 82-2-554-7200
Fax: 82-2-558-5932 or
82-2-558-5934

Malaysia - Kuala Lumpur
Tel: 60-3-6201-9857
Fax: 60-3-6201-9859

Malaysia - Penang
Tel: 60-4-227-8870
Fax: 60-4-227-4068

Philippines - Manila
Tel: 63-2-634-9065
Fax: 63-2-634-9069

Singapore
Tel: 65-6334-8870
Fax: 65-6334-8850

Taiwan - Hsin Chu
Tel: 886-3-5778-366
Fax: 886-3-5770-955

Taiwan - Kaohsiung
Tel: 886-7-536-4818
Fax: 886-7-330-9305

Taiwan - Taipei
Tel: 886-2-2500-6610
Fax: 886-2-2508-0102

Thailand - Bangkok
Tel: 66-2-694-1351
Fax: 66-2-694-1350

EUROPE

Austria - Wels
Tel: 43-7242-2244-39
Fax: 43-7242-2244-393

Denmark - Copenhagen
Tel: 45-4450-2828
Fax: 45-4485-2829

France - Paris
Tel: 33-1-69-53-63-20
Fax: 33-1-69-30-90-79

Germany - Munich
Tel: 49-89-627-144-0
Fax: 49-89-627-144-44

Italy - Milan
Tel: 39-0331-742611
Fax: 39-0331-466781

Netherlands - Drunen
Tel: 31-416-690399
Fax: 31-416-690340

Spain - Madrid
Tel: 34-91-708-08-90
Fax: 34-91-708-08-91

UK - Wokingham
Tel: 44-118-921-5869
Fax: 44-118-921-5820

08/02/11



Компания «ЭлектроПласт» предлагает заключение долгосрочных отношений при поставках импортных электронных компонентов на взаимовыгодных условиях!

Наши преимущества:

- Оперативные поставки широкого спектра электронных компонентов отечественного и импортного производства напрямую от производителей и с крупнейших мировых складов;
- Поставка более 17-ти миллионов наименований электронных компонентов;
- Поставка сложных, дефицитных, либо снятых с производства позиций;
- Оперативные сроки поставки под заказ (от 5 рабочих дней);
- Экспресс доставка в любую точку России;
- Техническая поддержка проекта, помощь в подборе аналогов, поставка прототипов;
- Система менеджмента качества сертифицирована по Международному стандарту ISO 9001;
- Лицензия ФСБ на осуществление работ с использованием сведений, составляющих государственную тайну;
- Поставка специализированных компонентов (Xilinx, Altera, Analog Devices, Intersil, Interpoint, Microsemi, Aeroflex, Peregrine, Syfer, Eurofarad, Texas Instrument, Miteq, Cobham, E2V, MA-COM, Hittite, Mini-Circuits, General Dynamics и др.);

Помимо этого, одним из направлений компании «ЭлектроПласт» является направление «Источники питания». Мы предлагаем Вам помощь Конструкторского отдела:

- Подбор оптимального решения, техническое обоснование при выборе компонента;
- Подбор аналогов;
- Консультации по применению компонента;
- Поставка образцов и прототипов;
- Техническая поддержка проекта;
- Защита от снятия компонента с производства.



Как с нами связаться

Телефон: 8 (812) 309 58 32 (многоканальный)

Факс: 8 (812) 320-02-42

Электронная почта: org@eplast1.ru

Адрес: 198099, г. Санкт-Петербург, ул. Калинина, дом 2, корпус 4, литера А.