Chipsmall Limited consists of a professional team with an average of over 10 year of expertise in the distribution of electronic components. Based in Hongkong, we have already established firm and mutual-benefit business relationships with customers from,Europe,America and south Asia,supplying obsolete and hard-to-find components to meet their specific needs.

With the principle of “Quality Parts,Customers Priority,Honest Operation,and Considerate Service”,our business mainly focus on the distribution of electronic components. Line cards we deal with include Microchip,ALPS,ROHM,Xilinx,Pulse,ON,Everlight and Freescale. Main products comprise IC,Modules,Potentiometer,IC Socket,Relay,Connector.Our parts cover such applications as commercial,industrial, and automotives areas.

We are looking forward to setting up business relationship with you and hope to provide you with the best service and solution. Let us make a better world for our industry!

Contact us

Tel: +86-755-8981 8866 Fax: +86-755-8427 6832
Email & Skype: info@chipsmall.com Web: www.chipsmall.com
Address: A1208, Overseas Decoration Building, #122 Zhenhua RD., Futian, Shenzhen, China

PIC16F610/16HV610
PIC16F616/16HV616
Data Sheet

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

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.

Trademarks

QUALITY MANAGEMENT SYSTEM
CERTIFIED BY DNV
ISO/TS 16949:2002

Microchip received ISO/TS-16949:2002 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.

PIC16F610/616/16HV610/616

14-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 oscillator/clock input
  - DC – 200 ns instruction cycle
- Interrupt Capability
- 8-Level Deep Hardware Stack
- Direct, Indirect and Relative Addressing modes

Special Microcontroller Features:

- Precision Internal Oscillator:
  - Factory calibrated to ±1%, typical
  - User selectable frequency: 4 MHz or 8 MHz
- Power-Saving Sleep mode
- Voltage Range:
  - PIC16F610/616: 2.0V to 5.5V
  - PIC16HV610/616: 2.0V to user defined maximum **(see note)**
- Industrial and Extended Temperature Range
- Power-on Reset (POR)
- Power-up Timer (PWRT) and Oscillator Start-up Timer (OST)
- Brown-out Reset (BOR)
- Watchdog Timer (WDT) with Independent Oscillator for Reliable Operation
- Multiplexed Master Clear with Pull-up/Input Pin
- Programmable Code Protection
- High Endurance Flash:
  - 100,000 write Flash endurance
  - Flash retention: > 40 years

Low-Power Features:

- Standby Current:
  - 50 nA @ 2.0V, typical
- Operating Current:
  - 20 μA @ 32 kHz, 2.0V, typical
  - 220 μA @ 4 MHz, 2.0V, typical
- Watchdog Timer Current:
  - 1 μA @ 2.0V, typical

Note: Voltage across internal shunt regulator cannot exceed 5V.

Peripheral Features:

- Shunt Voltage Regulator (PIC16HV610/616 only):
  - 5 volt regulation
  - 4 mA to 50 mA shunt range
- 11 I/O Pins and 1 Input Only
  - High current source/sink for direct LED drive
  - Interrupt-on-Change pins
  - Individually programmable weak pull-ups
- Analog Comparator module with:
  - Two analog comparators
  - Programmable on-chip voltage reference (CV_{REF}) module (% of V_{DD})
  - Fixed Voltage Reference
  - Comparator inputs and outputs externally accessible
  - SR Latch
  - Built-In Hysteresis (user selectable)
- Timer0: 8-Bit Timer/Counter with 8-Bit Programmable Prescaler
- Enhanced Timer1:
  - 16-bit timer/counter with prescaler
  - External Timer1 Gate (count enable)
  - Option to use OSC1 and OSC2 in LP mode as Timer1 oscillator if INTOSC mode selected
  - Timer1 oscillator
- In-Circuit Serial Programming™ (ICSP™) via Two Pins

PIC16F616/16HV616 only:

- A/D Converter:
  - 10-bit resolution
  - 8 external input channels
  - 2 internal reference channels
- Timer2: 8-Bit Timer/Counter with 8-Bit Period Register, Prescaler and Postscaler
- Enhanced Capture, Compare, PWM module:
  - 16-bit Capture, max. resolution 12.5 ns
  - 16-bit Compare, max. resolution 200 ns
  - 10-bit PWM with 1, 2 or 4 output channels, programmable “dead time”, max. frequency 20 kHz

Device	Program Memory	Data Memory	I/O	10-bit A/D (ch)	Comparators	Timers 8/16-bit	Voltage Range
	Flash (words)	SRAM (bytes)					
PIC16F610	1024	64	11	—	2	1/1	2.0-5.5V
PIC16HV610	1024	64	11	—	2	1/1	2.0-user defined
PIC16F616	2048	128	11	8	2	2/1	2.0-5.5V
PIC16HV616	2048	128	11	8	2	2/1	2.0-user defined

PIC16F610/16HV610 14-Pin Diagram (PDIP, SOIC, TSSOP)

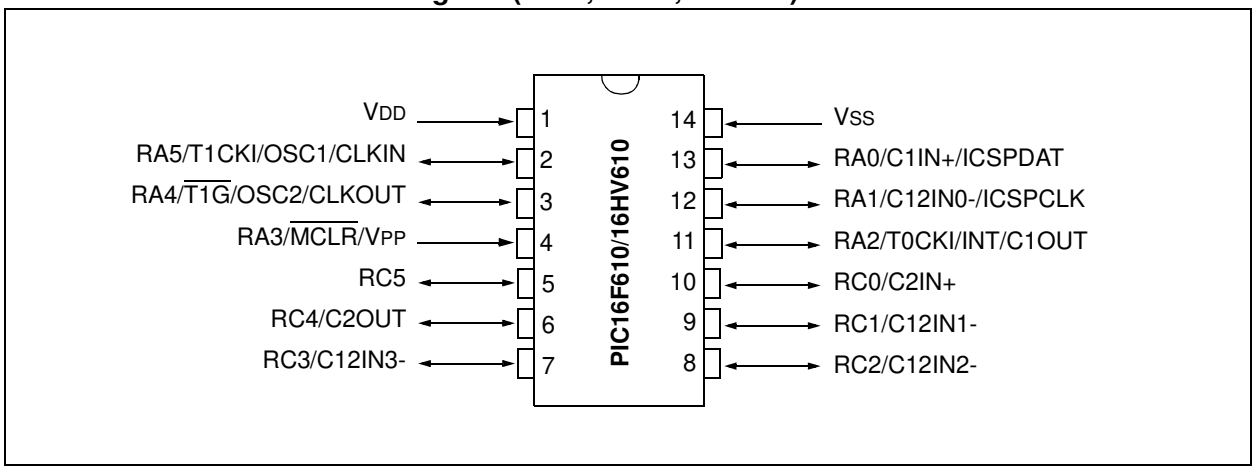


TABLE 1: PIC16F610/16HV610 14-PIN SUMMARY

I/O	Pin	Comparators	Timer	Interrupts	Pull-ups	Basic
RA0	13	C1IN+	—	IOC	Y	ICSPDAT
RA1	12	C12IN0-	—	IOC	Y	ICSPCLK
RA2	11	C1OUT	T0CKI	INT/IOC	Y	—
RA3(1)	4	—	—	IOC	Y(2)	MCLR/VPP
RA4	3	—	T1G	IOC	Y	OSC2/CLKOUT
RA5	2	—	T1CKI	IOC	Y	OSC1/CLKIN
RC0	10	C2IN+	—	—	—	—
RC1	9	C12IN1-	—	—	—	—
RC2	8	C12IN2-	—	—	—	—
RC3	7	C12IN3-	—	—	—	—
RC4	6	C2OUT	—	—	—	—
RC5	5	—	—	—	—	—
—	1	—	—	—	—	VDD
—	14	—	—	—	—	VSS

Note 1: Input only.

2: Only when pin is configured for external MCLR.

PIC16F616/16HV616 14-Pin Diagram (PDIP, SOIC, TSSOP)

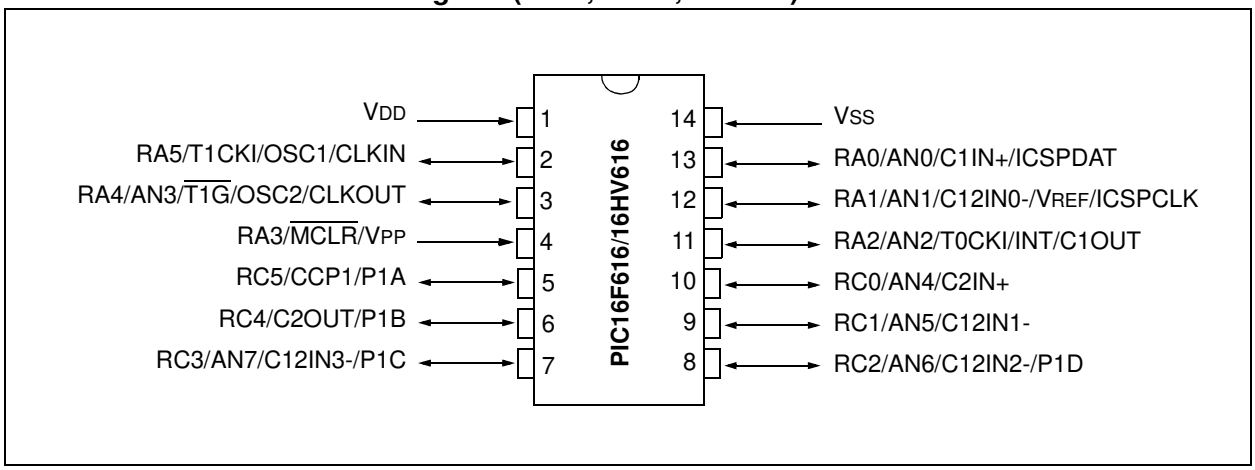


TABLE 2: PIC16F616/16HV616 14-PIN SUMMARY

I/O	Pin	Analog	Comparators	Timer	CCP	Interrupts	Pull-ups	Basic
RA0	13	AN0	C1IN+	—	—	IOC	Y	ICSPDAT
RA1	12	AN1/VREF	C12IN0-	—	—	IOC	Y	ICSPCLK
RA2	11	AN2	C1OUT	T0CKI	—	INT/IOC	Y	—
RA3(1)	4	—	—	—	—	IOC	Y(2)	MCLR/VPP
RA4	3	AN3	—	T1G	—	IOC	Y	OSC2/CLKOUT
RA5	2	—	—	T1CKI	—	IOC	Y	OSC1/CLKIN
RC0	10	AN4	C2IN+	—	—	—	—	—
RC1	9	AN5	C12IN1-	—	—	—	—	—
RC2	8	AN6	C12IN2-	—	P1D	—	—	—
RC3	7	AN7	C12IN3-	—	P1C	—	—	—
RC4	6	—	C2OUT	—	P1B	—	—	—
RC5	5	—	—	—	CCP1/P1A	—	—	—
—	1	—	—	—	—	—	—	VDD
—	14	—	—	—	—	—	—	VSS

Note 1: Input only.

2: Only when pin is configured for external MCLR.

PIC16F610/16HV610 16-Pin Diagram (QFN)

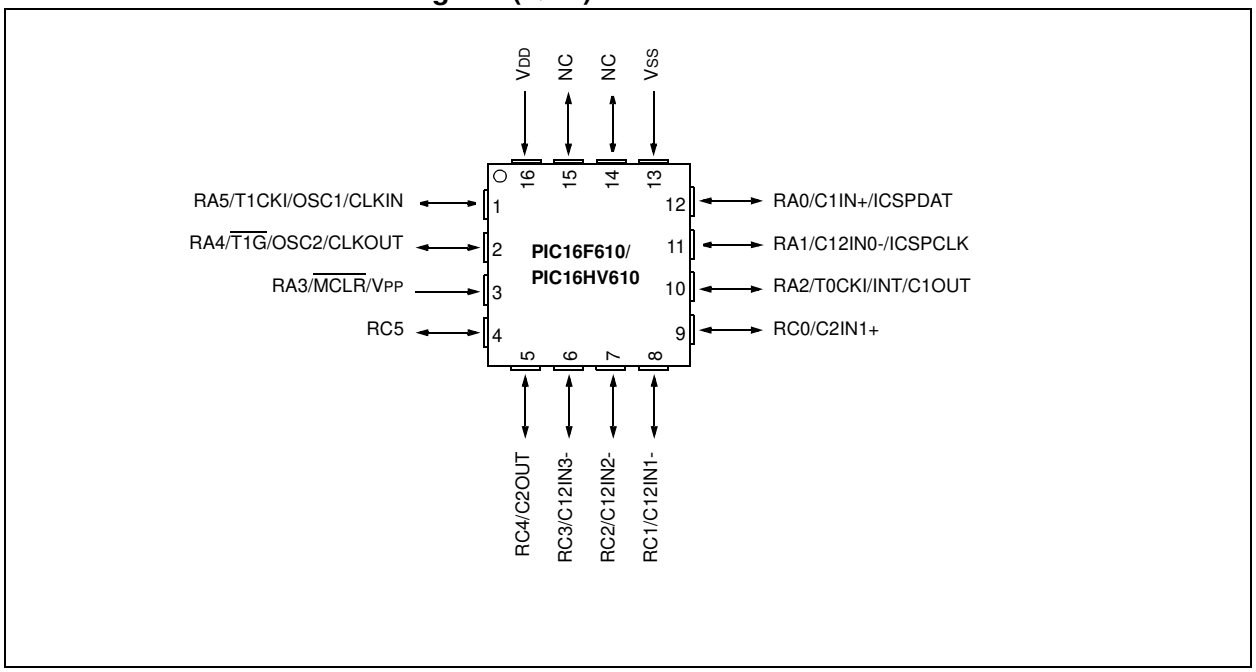


TABLE 3: PIC16F610/16HV610 16-PIN SUMMARY

I/O	Pin	Comparators	Timers	Interrupts	Pull-ups	Basic
RA0	12	C1IN+	—	IOC	Y	ICSPDAT
RA1	11	C12IN0-	—	IOC	Y	ICSPCLK
RA2	10	C1OUT	T0CKI	INT/IOC	Y	—
RA3(1)	3	—	—	IOC	Y(2)	$\overline{MCLR}$/VPP
RA4	2	—	$\overline{T1G}$	IOC	Y	OSC2/CLKOUT
RA5	1	—	T1CKI	IOC	Y	OSC1/CLKIN
RC0	9	C2IN+	—	—	—	—
RC1	8	C12IN1-	—	—	—	—
RC2	7	C12IN2-	—	—	—	—
RC3	6	C12IN3-	—	—	—	—
RC4	5	C2OUT	—	—	—	—
RC5	4	—	—	—	—	—
—	16	—	—	—	—	VDD
—	13	—	—	—	—	VSS

Note 1: Input only.

2: Only when pin is configured for external $\overline{MCLR}$.

PIC16F616/16HV616 16-Pin Diagram (QFN)

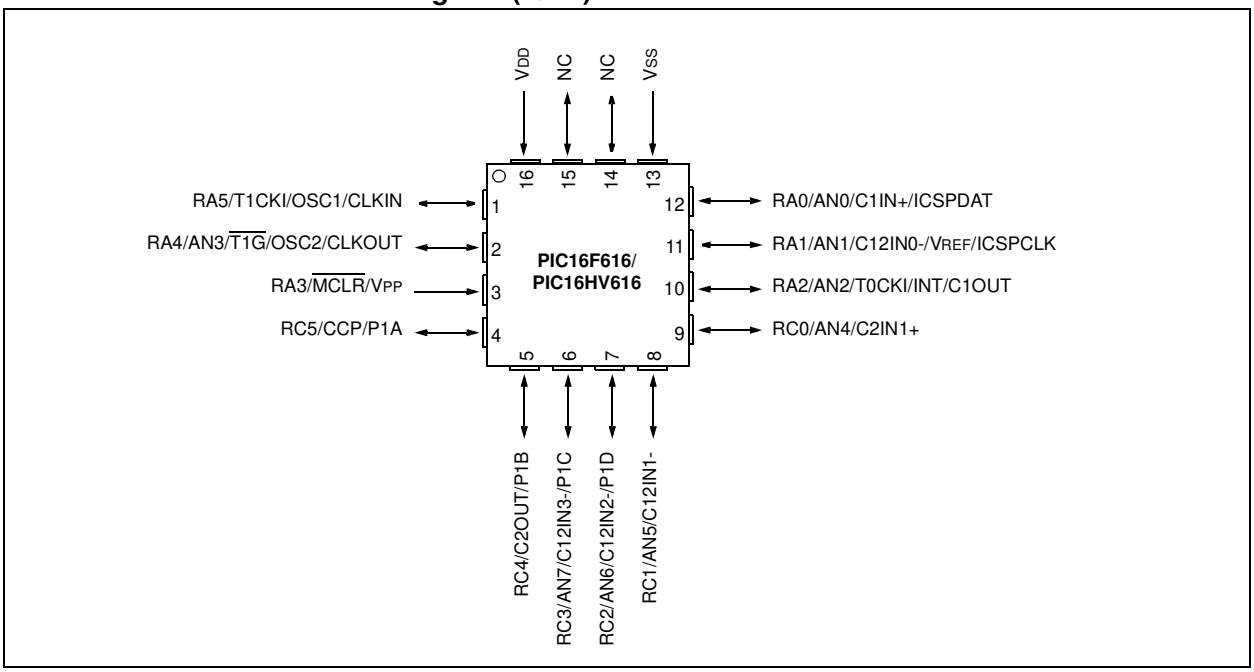


TABLE 4: PIC16F616/16HV616 16-PIN SUMMARY

I/O	Pin	Analog	Comparators	Timers	CCP	Interrupts	Pull-ups	Basic
RA0	12	AN0	C1IN+	—	—	IOC	Y	ICSPDAT
RA1	11	AN1/VREF	C12IN0-	—	—	IOC	Y	ICSPCLK
RA2	10	AN2	C1OUT	T0CKI	—	INT/IOC	Y	—
RA3(1)	3	—	—	—	—	IOC	Y(2)	$\overline{MCLR}$/VPP
RA4	2	AN3	—	$\overline{T1G}$	—	IOC	Y	OSC2/CLKOUT
RA5	1	—	—	T1CKI	—	IOC	Y	OSC1/CLKIN
RC0	9	AN4	C2IN+	—	—	—	—	—
RC1	8	AN5	C12IN1-	—	—	—	—	—
RC2	7	AN6	C12IN2-	—	P1D	—	—	—
RC3	6	AN7	C12IN3-	—	P1C	—	—	—
RC4	5	—	C2OUT	—	P1B	—	—	—
RC5	4	—	—	—	CCP1/P1A	—	—	—
—	16	—	—	—	—	—	—	VDD
—	13	—	—	—	—	—	—	VSS

Note 1: Input only.

2: Only when pin is configured for external $\overline{MCLR}$.

Table of Contents

TO OUR VALUED CUSTOMERS

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

If you have any questions or comments regarding this publication, please contact the Marketing Communications Department via E-mail at **docerrors@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.

1.0 DEVICE OVERVIEW

The PIC16F610/616/16HV610/616 is covered by this data sheet. It is available in 14-pin PDIP, SOIC, TSSOP and 16-pin QFN packages.

Block Diagrams and pinout descriptions of the devices are as follows:

- PIC16F610/16HV610 (Figure 1-1, Table 1-1)
- PIC16F616/16HV616 (Figure 1-2, Table 1-2)

FIGURE 1-1: PIC16F610/16HV610 BLOCK DIAGRAM

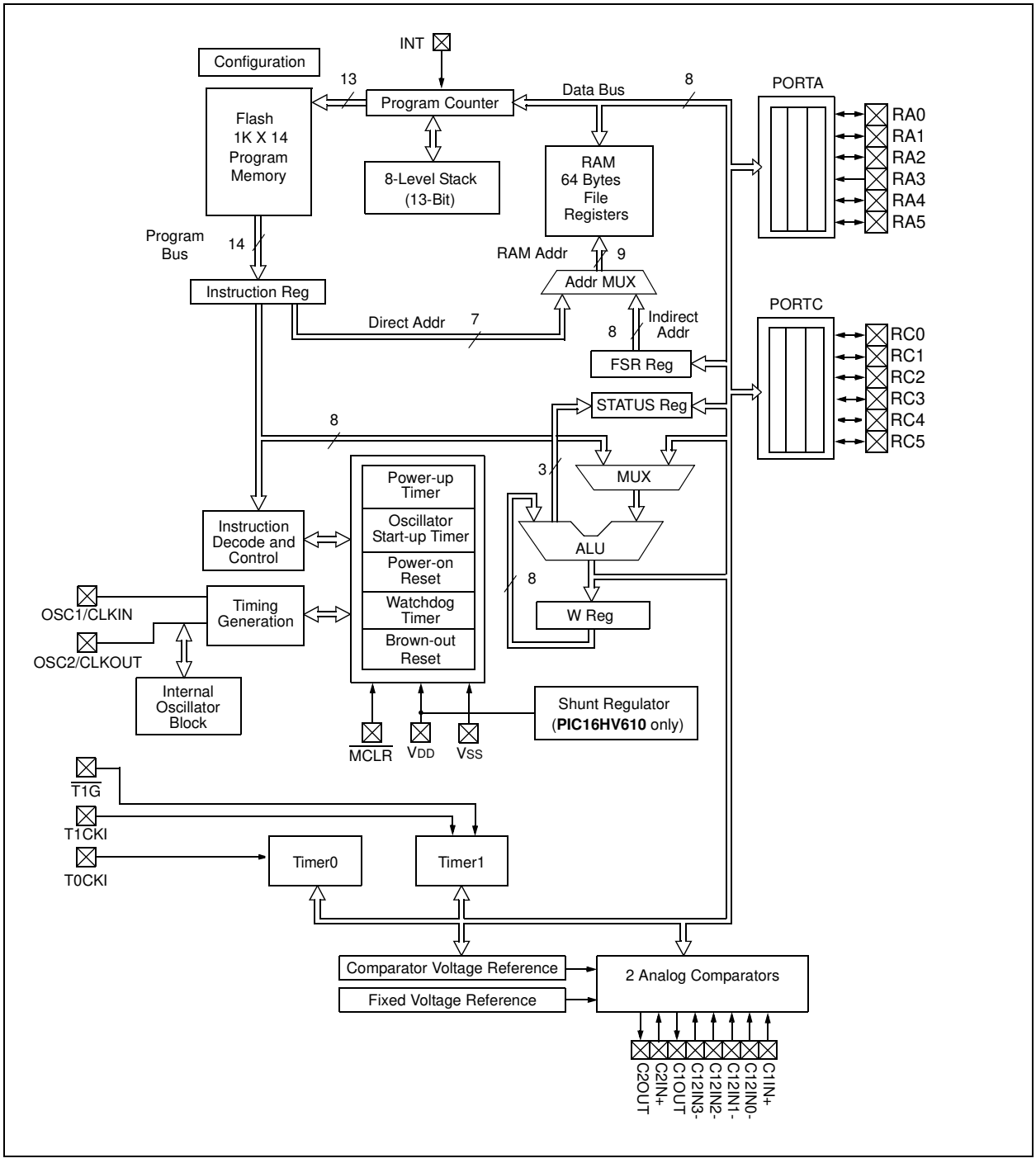


FIGURE 1-2: PIC16F616/16HV616 BLOCK DIAGRAM

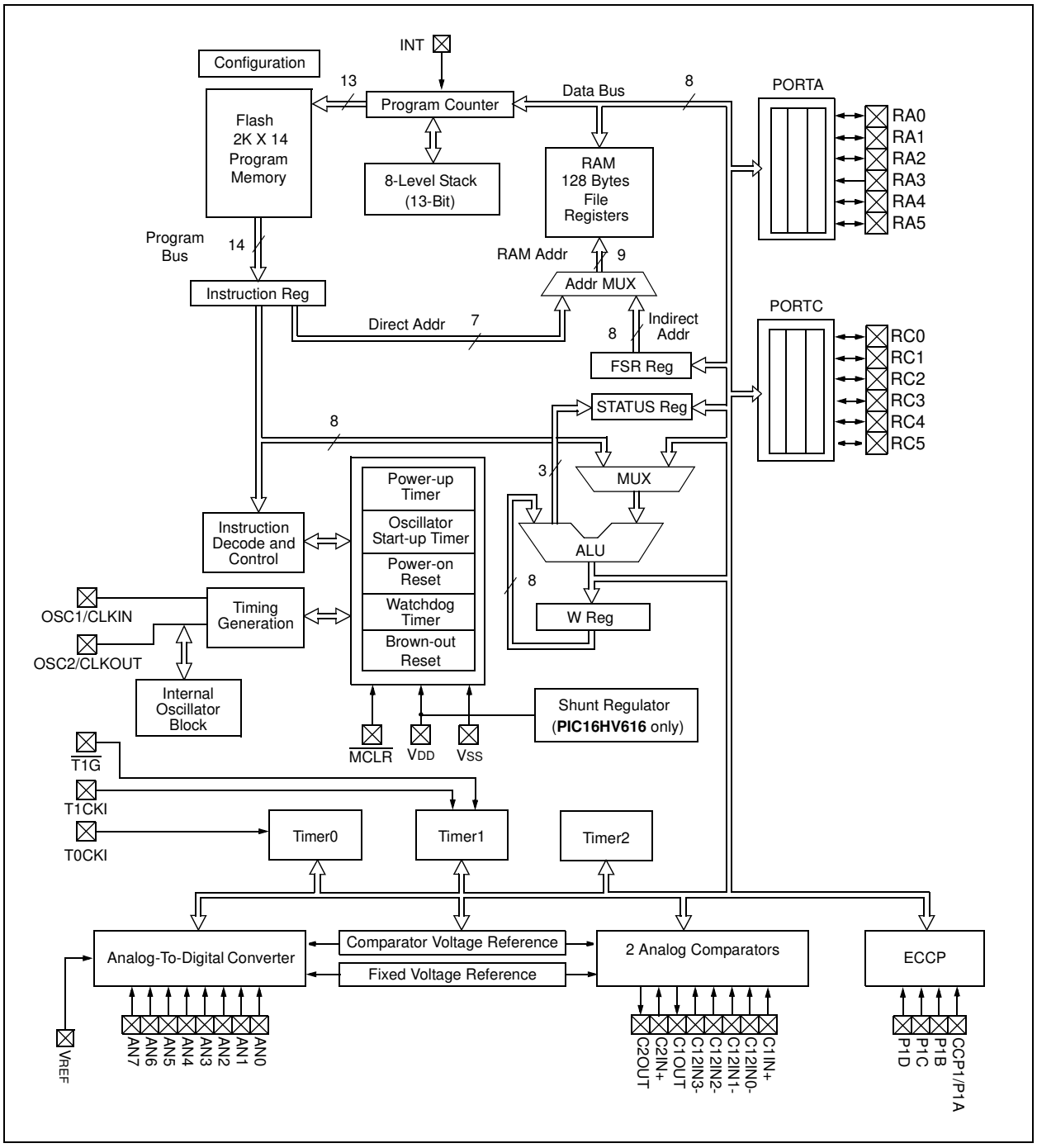


TABLE 1-1: PIC16F610/16HV610 PINOUT DESCRIPTION

Name	Function	Input Type	Output Type	Description
RA0/C1IN+/ICSPDAT	RA0	TTL	CMOS	PORTA I/O with prog. pull-up and interrupt-on-change
	C1IN+	AN	—	Comparator C1 non-inverting input
	ICSPDAT	ST	CMOS	Serial Programming Data I/O
RA1/C12IN0-/ICSPCLK	RA1	TTL	CMOS	PORTA I/O with prog. pull-up and interrupt-on-change
	C12IN0-	AN	—	Comparators C1 and C2 inverting input
	ICSPCLK	ST	—	Serial Programming Clock
RA2/T0CKI/INT/C1OUT	RA2	ST	CMOS	PORTA I/O with prog. pull-up and interrupt-on-change
	T0CKI	ST	—	Timer0 clock input
	INT	ST	—	External Interrupt
	C1OUT	—	CMOS	Comparator C1 output
RA3/$\overline{MCLR}$/VPP	RA3	TTL	—	PORTA input with interrupt-on-change
	$\overline{MCLR}$	ST	—	Master Clear w/internal pull-up
	VPP	HV	—	Programming voltage
RA4/$\overline{T1G}$/OSC2/CLKOUT	RA4	TTL	CMOS	PORTA I/O with prog. pull-up and interrupt-on-change
	$\overline{T1G}$	ST	—	Timer1 gate (count enable)
	OSC2	—	XTAL	Crystal/Resonator
	CLKOUT	—	CMOS	FOSC/4 output
RA5/T1CKI/OSC1/CLKIN	RA5	TTL	CMOS	PORTA I/O with prog. pull-up and interrupt-on-change
	T1CKI	ST	—	Timer1 clock input
	OSC1	XTAL	—	Crystal/Resonator
	CLKIN	ST	—	External clock input/RC oscillator connection
RC0/C2IN+	RC0	TTL	CMOS	PORTC I/O
	C2IN+	AN	—	Comparator C2 non-inverting input
RC1/C12IN1-	RC1	TTL	CMOS	PORTC I/O
	C12IN1-	AN	—	Comparators C1 and C2 inverting input
RC2/C12IN2-	RC2	TTL	CMOS	PORTC I/O
	C12IN2-	AN	—	Comparators C1 and C2 inverting input
RC3/C12IN3-	RC3	TTL	CMOS	PORTC I/O
	C12IN3-	AN	—	Comparators C1 and C2 inverting input
RC4/C2OUT	RC4	TTL	CMOS	PORTC I/O
	C2OUT	—	CMOS	Comparator C2 output
RC5	RC5	TTL	CMOS	PORTC I/O
VDD	VDD	Power	—	Positive supply
VSS	VSS	Power	—	Ground reference

Legend: AN = Analog input or output; CMOS = CMOS compatible input or output; HV = High Voltage
ST = Schmitt Trigger input with CMOS levels; TTL = TTL compatible input; XTAL = Crystal

TABLE 1-2: PIC16F616/16HV616 PINOUT DESCRIPTION

Name	Function	Input Type	Output Type	Description
RA0/AN0/C1IN+/ICSPDAT	RA0	TTL	CMOS	PORTA I/O with prog. pull-up and interrupt-on-change
	AN0	AN	—	A/D Channel 0 input
	C1IN+	AN	—	Comparator C1 non-inverting input
	ICSPDAT	ST	CMOS	Serial Programming Data I/O
RA1/AN1/C12IN0-/VREF/ICSPCLK	RA1	TTL	CMOS	PORTA I/O with prog. pull-up and interrupt-on-change
	AN1	AN	—	A/D Channel 1 input
	C12IN0-	AN	—	Comparators C1 and C2 inverting input
	VREF	AN	—	External Voltage Reference for A/D
	ICSPCLK	ST	—	Serial Programming Clock
RA2/AN2/T0CKI/INT/C1OUT	RA2	ST	CMOS	PORTA I/O with prog. pull-up and interrupt-on-change
	AN2	AN	—	A/D Channel 2 input
	T0CKI	ST	—	Timer0 clock input
	INT	ST	—	External Interrupt
	C1OUT	—	CMOS	Comparator C1 output
RA3/$\overline{\text{MCLR}}$/VPP	RA3	TTL	—	PORTA input with interrupt-on-change
	$\overline{\text{MCLR}}$	ST	—	Master Clear w/internal pull-up
	VPP	HV	—	Programming voltage
RA4/AN3/$\overline{\text{T1G}}$/OSC2/CLKOUT	RA4	TTL	CMOS	PORTA I/O with prog. pull-up and interrupt-on-change
	AN3	AN	—	A/D Channel 3 input
	$\overline{\text{T1G}}$	ST	—	Timer1 gate (count enable)
	OSC2	—	XTAL	Crystal/Resonator
	CLKOUT	—	CMOS	FOSC/4 output
RA5/T1CKI/OSC1/CLKIN	RA5	TTL	CMOS	PORTA I/O with prog. pull-up and interrupt-on-change
	T1CKI	ST	—	Timer1 clock input
	OSC1	XTAL	—	Crystal/Resonator
	CLKIN	ST	—	External clock input/RC oscillator connection
RC0/AN4/C2IN+	RC0	TTL	CMOS	PORTC I/O
	AN4	AN	—	A/D Channel 4 input
	C2IN+	AN	—	Comparator C2 non-inverting input
RC1/AN5/C12IN1-	RC1	TTL	CMOS	PORTC I/O
	AN5	AN	—	A/D Channel 5 input
	C12IN1-	AN	—	Comparators C1 and C2 inverting input
RC2/AN6/C12IN2-/P1D	RC2	TTL	CMOS	PORTC I/O
	AN6	AN	—	A/D Channel 6 input
	C12IN2-	AN	—	Comparators C1 and C2 inverting input
	P1D	—	CMOS	PWM output
RC3/AN7/C12IN3-/P1C	RC3	TTL	CMOS	PORTC I/O
	AN7	AN	—	A/D Channel 7 input
	C12IN3-	AN	—	Comparators C1 and C2 inverting input
	P1C	—	CMOS	PWM output
RC4/C2OUT/P1B	RC4	TTL	CMOS	PORTC I/O
	C2OUT	—	CMOS	Comparator C2 output
	P1B	—	CMOS	PWM output
RC5/CCP1/P1A	RC5	TTL	CMOS	PORTC I/O
	CCP1	ST	CMOS	Capture input/Compare output
	P1A	—	CMOS	PWM output
VDD	VDD	Power	—	Positive supply
VSS	VSS	Power	—	Ground reference

Legend: AN = Analog input or output; CMOS = CMOS compatible input or output; HV = High Voltage
ST = Schmitt Trigger input with CMOS levels; TTL = TTL compatible input; XTAL = Crystal

2.0 MEMORY ORGANIZATION

2.1 Program Memory Organization

The PIC16F610/616/16HV610/616 has a 13-bit program counter capable of addressing an 8K x 14 program memory space. Only the first 1K x 14 (0000h-3FF) for the PIC16F610/16HV610 and the first 2K x 14 (0000h-07FFh) for the PIC16F616/16HV616 is physically implemented. Accessing a location above these boundaries will cause a wraparound within the first 1K x 14 space (PIC16F610/16HV610) and 2K x 14 space (PIC16F616/16HV616). 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 PIC16F610/16HV610

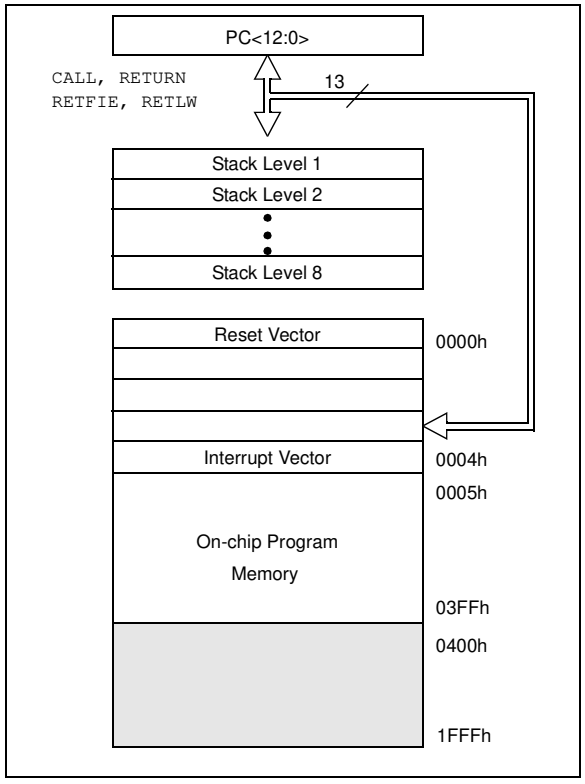
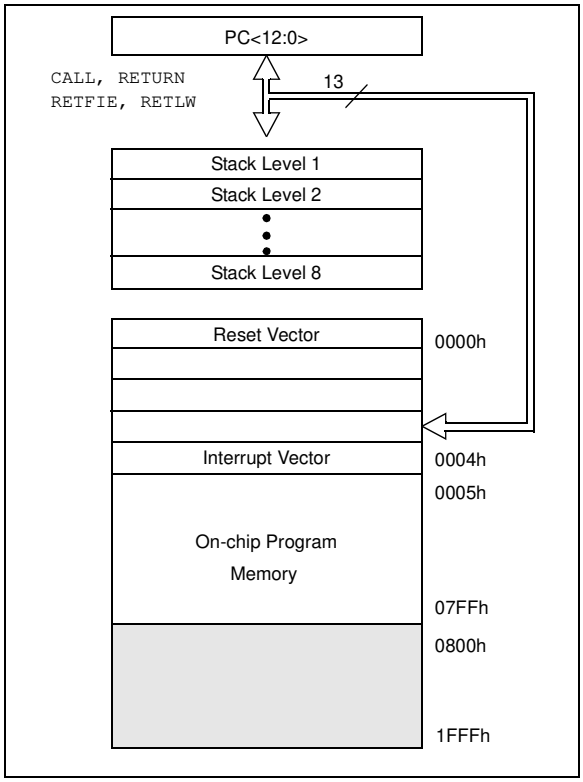


FIGURE 2-2: PROGRAM MEMORY MAP AND STACK FOR THE PIC16F616/16HV616



2.2 Data Memory Organization

The data memory (see Figure 2-4) is partitioned into two 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. PIC16F610/16HV610 Register locations 40h-7Fh in Bank 0 are General Purpose Registers, implemented as static RAM. PIC16F616/16HV616 Register locations 20h-7Fh in Bank 0 and A0h-BFh in Bank 1 are General Purpose Registers, implemented as static RAM. Register locations F0h-FFh in Bank 1 point to addresses 70h-7Fh in Bank 0. All other RAM is unimplemented and returns '0' when read. The RP0 bit of the STATUS register is the bank select bit.

RP0

0 → Bank 0 is selected

1 → Bank 1 is selected

Note:	The IRP and RP1 bits of the STATUS register are reserved and should always be maintained as '0's.

2.2.1 GENERAL PURPOSE REGISTER FILE

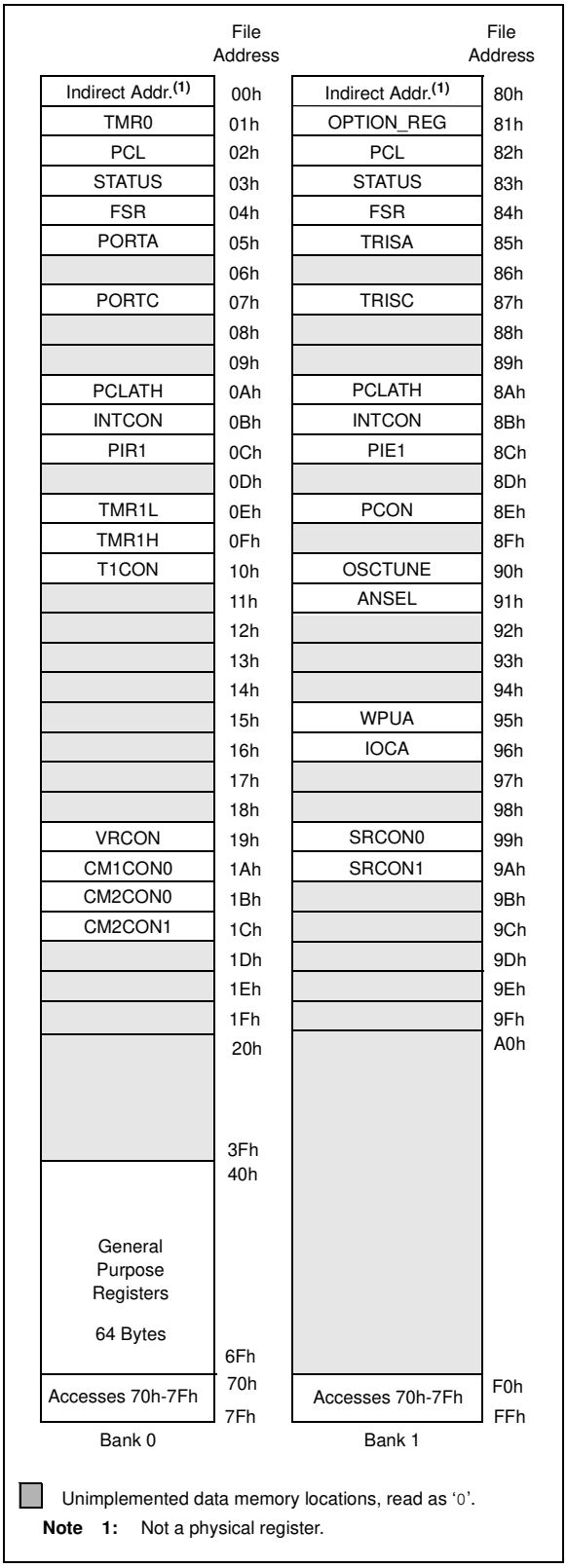
The register file is organized as 64 x 8 in the PIC16F610/16HV610 and 128 x 8 in the PIC16F616/16HV616. Each register is accessed, either directly or indirectly, through the File Select Register (FSR) (see **Section 2.4 "Indirect Addressing, INDF and FSR Registers"**).

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.

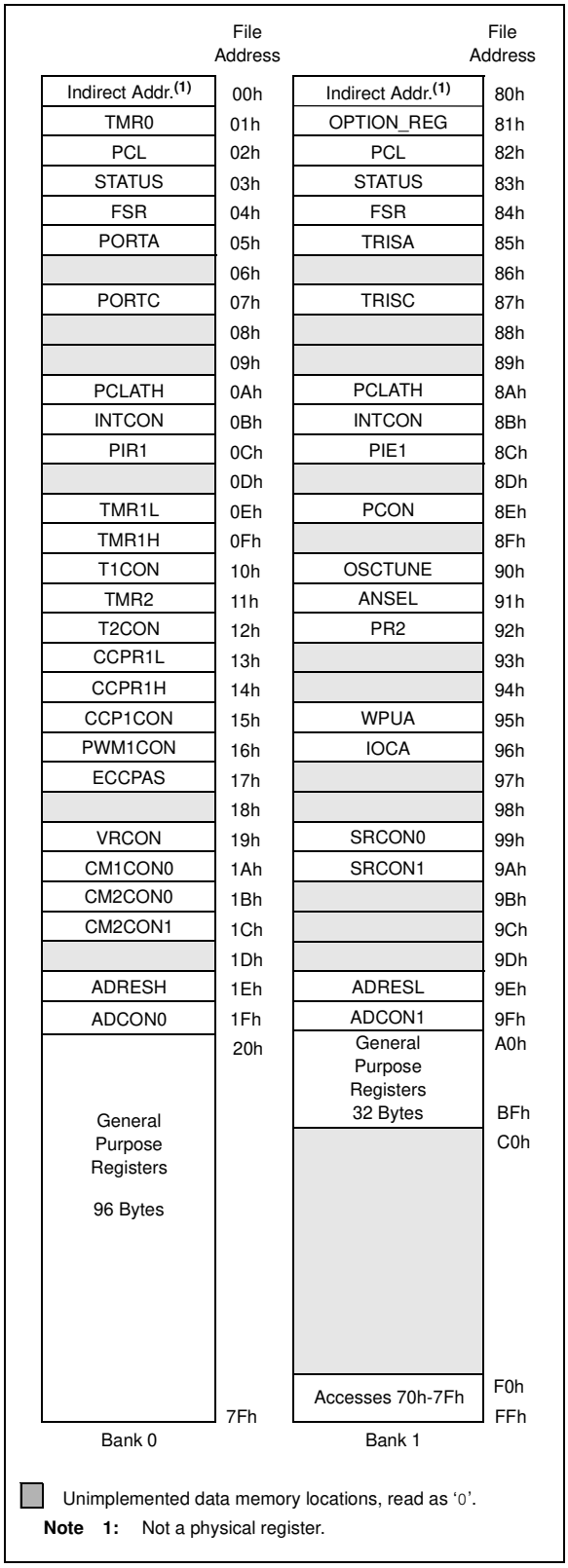
FIGURE 2-3: DATA MEMORY MAP OF THE PIC16F610/16HV610



Unimplemented data memory locations, read as '0'.

Note 1: Not a physical register.

FIGURE 2-4: DATA MEMORY MAP OF THE PIC16F616/16HV616



Unimplemented data memory locations, read as '0'.

Note 1: Not a physical register.

TABLE 2-1: PIC16F610/616/16HV610/616 SPECIAL FUNCTION REGISTERS SUMMARY BANK 0

Addr	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on POR, BOR	Page
Bank 0											
00h	INDF	Addressing this location uses contents of FSR to address data memory (not a physical register)								xxxx xxxx	24, 116
01h	TMR0	Timer0 Module's Register								xxxx xxxx	45, 116
02h	PCL	Program Counter's (PC) Least Significant Byte								0000 0000	24, 116
03h	STATUS	IRP(1)	RP1(1)	RP0	$\overline{TO}$	$\overline{PD}$	Z	DC	C	0001 1xxx	18, 116
04h	FSR	Indirect Data Memory Address Pointer								xxxx xxxx	24, 116
05h	PORTA	—	—	RA5	RA4	RA3	RA2	RA1	RA0	--x0 x000	33, 116
06h	—	Unimplemented								—	—
07h	PORTC	—	—	RC5	RC4	RC3	RC2	RC1	RC0	--xx 00xx	42, 116
08h	—	Unimplemented								—	—
09h	—	Unimplemented								—	—
0Ah	PCLATH	—	—	—	Write Buffer for upper 5 bits of Program Counter					---0 0000	24, 116
0Bh	INTCON	GIE	PEIE	T0IE	INTE	RAIE	T0IF	INTF	RAIF	0000 0000	20, 116
0Ch	PIR1	—	ADIF(2)	CCP1IF(2)	C2IF	C1IF	—	TMR2IF(2)	TMR1IF	-000 0-00	22, 116
0Dh	—	Unimplemented								—	—
0Eh	TMR1L	Holding Register for the Least Significant Byte of the 16-bit TMR1 Register								xxxx xxxx	49, 116
0Fh	TMR1H	Holding Register for the Most Significant Byte of the 16-bit TMR1 Register								xxxx xxxx	49, 116
10h	T1CON	T1GINV	TMR1GE	T1CKPS1	T1CKPS0	T1OSCEN	$\overline{T1SYNC}$	TMR1CS	TMR1ON	0000 0000	52, 116
11h	TMR2(2)	Timer2 Module Register								0000 0000	55, 116
12h	T2CON(2)	—	TOUTPS3	TOUTPS2	TOUTPS1	TOUTPS0	TMR2ON	T2CKPS1	T2CKPS0	-000 0000	56, 116
13h	CCPR1L(2)	Capture/Compare/PWM Register 1 Low Byte								xxxx xxxx	86, 116
14h	CCPR1H(2)	Capture/Compare/PWM Register 1 High Byte								xxxx xxxx	86, 116
15h	CCP1CON(2)	P1M1	P1M0	DC1B1	DC1B0	CCP1M3	CCP1M2	CCP1M1	CCP1M0	0000 0000	85, 116
16h	PWM1CON(2)	PRSEN	PDC6	PDC5	PDC4	PDC3	PDC2	PDC1	PDC0	0000 0000	85, 116
17h	ECCPAS(2)	ECCPASE	ECCPAS2	ECCPAS1	ECCPAS0	PSSAC1	PSSAC0	PSSBD1	PSSBD0	0000 0000	102, 116
18h	—	Unimplemented								—	—
19h	VRCON	C1VREN	C2VREN	VRR	FVREN	VR3	VR2	VR1	VR0	0000 0000	72, 116
1Ah	CM1CON0	C1ON	C1OUT	C1OE	C1POL	—	C1R	C1CH1	C1CH0	0000 -000	62, 116
1Bh	CM2CON0	C2ON	C2OUT	C2OE	C2POL	—	C2R	C2CH1	C2CH0	0000 -000	63, 116
1Ch	CM2CON1	MC1OUT	MC2OUT	—	T1ACS	C1HYS	C2HYS	T1GSS	C2SYNC	00-0 0010	65, 116
1Dh	—	Unimplemented								—	—
1Eh	ADRESH(2,3)	Most Significant 8 bits of the left shifted A/D result or 2 bits of right shifted result								xxxx xxxx	80, 116
1Fh	ADCON0(2)	ADFM	VCFG	CHS3	CHS2	CHS1	CHS0	GO/$\overline{DONE}$	ADON	0000 0000	78, 116

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

Note 1: IRP and RP1 bits are reserved, always maintain these bits clear.

2: PIC16F616/16HV616 only.

3: Read-only register.

TABLE 2-2: PIC16F610/616/16HV610/616 SPECIAL FUNCTION 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	Page
Bank 1											
80h	INDF	Addressing this location uses contents of FSR to address data memory (not a physical register)								xxxx xxxx	24, 116
81h	OPTION_REG	$\overline{RAPU}$	INTEDG	T0CS	T0SE	PSA	PS2	PS1	PS0	1111 1111	19, 116
82h	PCL	Program Counter's (PC) Least Significant Byte								0000 0000	24, 116
83h	STATUS	IRP(1)	RP1(1)	RP0	$\overline{TO}$	$\overline{PD}$	Z	DC	C	0001 1xxx	18, 116
84h	FSR	Indirect Data Memory Address Pointer								xxxx xxxx	24, 116
85h	TRISA	—	—	TRISA5	TRISA4	TRISA3	TRISA2	TRISA1	TRISA0	--11 1111	33, 116
86h	—	Unimplemented								—	—
87h	TRISC	—	—	TRISC5	TRISC4	TRISC3	TRISC2	TRISC1	TRISC0	--11 1111	42, 116
88h	—	Unimplemented								—	—
89h	—	Unimplemented								—	—
8Ah	PCLATH	—	—	—	Write Buffer for upper 5 bits of Program Counter					---0 0000	24, 116
8Bh	INTCON	GIE	PEIE	T0IE	INTE	RAIE	T0IF	INTF	RAIF	0000 0000	20, 116
8Ch	PIE1	—	ADIE(3)	CCP1IE(3)	C2IE	C1IE	—	TMR2IE(3)	TMR1IE	-000 0-00	21, 116
8Dh	—	Unimplemented								—	—
8Eh	PCON	—	—	—	—	—	—	$\overline{POR}$	$\overline{BOR}$	---- --qq	23, 116
8Fh	—	Unimplemented								—	—
90h	OSCTUNE	—	—	—	TUN4	TUN3	TUN2	TUN1	TUN0	---0 0000	31, 117
91h	ANSEL	ANS7	ANS6	ANS5	ANS4	ANS3(3)	ANS2(3)	ANS1	ANS0	1111 1111	34, 117
92h	PR2(3)	Timer2 Module Period Register								1111 1111	55, 117
93h	—	Unimplemented								—	—
94h	—	Unimplemented								—	—
95h	WPUA	—	—	WPUA5	WPUA4	—	WPUA2	WPUA1	WPUA0	--11 -111	35, 117
96h	IOCA	—	—	IOCA5	IOCA4	IOCA3	IOCA2	IOCA1	IOCA0	--00 0000	35, 117
97h	—	Unimplemented								—	—
98h	—	Unimplemented								—	—
99h	SRCON0	SR1	SR0	C1SEN	C2REN	PULSS	PULSR	—	SRCLKEN	0000 00-0	69, 117
9Ah	SRCON1	SRCS1	SRCS0	—	—	—	—	—	—	00-- ----	69, 117
9Bh	—	Unimplemented								—	—
9Ch	—	Unimplemented								—	—
9Dh	—	Unimplemented								—	—
9Eh	ADRESL(3,4)	Least Significant 2 bits of the left shifted result or 8 bits of the right shifted result								xxxx xxxx	80, 117
9Fh	ADCON1(3)	—	ADCS2	ADCS1	ADCS0	—	—	—	—	-000 ----	79, 117

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

Note 1: IRP and RP1 bits are reserved, always maintain these bits clear.

2: RA3 pull-up is enabled when MCLRE is '1' in the Configuration Word register.

3: PIC16F616/16HV616 only.

4: Read-only Register.

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 $\overline{TO}$ and $\overline{PD}$ bits are not writable. Therefore, the result of an instruction with the STATUS register as destination may be different than intended.

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

It is recommended, therefore, that only `BCF`, `BSF`, `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 13.0 "Instruction Set Summary"**.

Note 1: Bits IRP and RP1 of the STATUS register are not used by the PIC16F610/616/16HV610/616 and should be maintained as clear. Use of these bits is not recommended, since this may affect upward compatibility with future products.

2: The C and DC bits operate as a $\overline{Borrow}$ and Digit $\overline{Borrow}$ out bit, respectively, in subtraction. See the `SUBLW` and `SUBWF` instructions for examples.

REGISTER 2-1: STATUS: STATUS REGISTER

Reserved	Reserved	R/W-0	R-1	R-1	R/W-x	R/W-x	R/W-x
IRP	RP1	RP0	$\overline{TO}$	$\overline{PD}$	Z	DC	C
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:** This bit is reserved and should be maintained as `0`

bit 6 **RP1:** This bit is reserved and should be maintained as `0`

bit 5 **RP0:** Register Bank Select bit (used for direct addressing)
`1` = Bank 1 (80h – FFh)
`0` = Bank 0 (00h – 7Fh)

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

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

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

bit 1 **DC:** Digit Carry/$\overline{Borrow}$ bit (`ADDWF`, `ADDLW`, `SUBLW`, `SUBWF` instructions), For $\overline{Borrow}$, 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/$\overline{Borrow}$ bit(1) (`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: For $\overline{Borrow}$, 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 5.1.3 "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
$\overline{RAPU}$	INTEDG	T0CS	T0SE	PSA	PS2	PS1	PS0
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 **$\overline{RAPU}$:** 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 RA2/INT pin
0 = Interrupt on falling edge of RA2/INT pin

bit 5 **T0CS:** Timer0 Clock Source Select bit
1 = Transition on RA2/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 RA2/T0CKI pin
0 = Increment on low-to-high transition on RA2/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

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, PORTA 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	RAIE	T0IF	INTF	RAIF
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 **RAIE:** PORTA Change Interrupt Enable bit**(1)**
1 = Enables the PORTA change interrupt
0 = Disables the PORTA 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 **RAIF:** PORTA Change Interrupt Flag bit
1 = When at least one of the PORTA <5:0> pins changed state (must be cleared in software)
0 = None of the PORTA <5:0> pins have changed state

Note 1: IOCA register must also be enabled.
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

U-0	R/W-0	R/W-0	R/W-0	R/W-0	U-0	R/W-0	R/W-0
—	ADIE(1)	CCP1IE(1)	C2IE	C1IE	—	TMR2IE(1)	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 **Unimplemented:** Read as '0'

bit 6 **ADIE:** A/D Converter (ADC) Interrupt Enable bit(1)
- 1 = Enables the ADC interrupt
- 0 = Disables the ADC interrupt

bit 5 **CCP1IE:** CCP1 Interrupt Enable bit(1)
- 1 = Enables the CCP1 interrupt
- 0 = Disables the CCP1 interrupt

bit 4 **C2IE:** Comparator C2 Interrupt Enable bit
- 1 = Enables the Comparator C2 interrupt
- 0 = Disables the Comparator C2 interrupt

bit 3 **C1IE:** Comparator C1 Interrupt Enable bit
- 1 = Enables the Comparator C1 interrupt
- 0 = Disables the Comparator C1 interrupt

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

bit 1 **TMR2IE:** Timer2 to PR2 Match Interrupt Enable bit(1)
- 1 = Enables the Timer2 to PR2 match interrupt
- 0 = Disables the Timer2 to PR2 match interrupt

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

Note 1: PIC16F616/16HV616 only. PIC16F610/16HV610 unimplemented, read as '0'.

2.2.2.5 PIR1 Register

The PIR1 register contains the peripheral interrupt flag bits, as shown in Register 2-5.

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 2-5: PIR1: PERIPHERAL INTERRUPT REQUEST REGISTER 1

U-0	R/W-0	R/W-0	R/W-0	R/W-0	U-0	R/W-0	R/W-0
—	ADIF(1)	CCP1IF(1)	C2IF	C1IF	—	TMR2IF(1)	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 **Unimplemented:** Read as '0'

bit 6 **ADIF:** A/D Interrupt Flag bit(1)
1 = A/D conversion complete
0 = A/D conversion has not completed or has not been started

bit 5 **CCP1IF:** CCP1 Interrupt Flag bit(1)
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

bit 4 **C2IF:** Comparator C2 Interrupt Flag bit
1 = Comparator C2 output has changed (must be cleared in software)
0 = Comparator C2 output has not changed

bit 3 **C1IF:** Comparator C1 Interrupt Flag bit
1 = Comparator C1 output has changed (must be cleared in software)
0 = Comparator C1 output has not changed

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

bit 1 **TMR2IF:** Timer2 to PR2 Match Interrupt Flag bit(1)
1 = Timer2 to PR2 match occurred (must be cleared in software)
0 = Timer2 to PR2 match has not occurred

bit 0 **TMR1IF:** Timer1 Overflow Interrupt Flag bit
1 = Timer1 register overflowed (must be cleared in software)
0 = Timer1 has not overflowed

Note 1: PIC16F616/16HV616 only. PIC16F610/16HV610 unimplemented, read as '0'.

2.2.2.6 PCON Register

The Power Control (PCON) register (see Table 12-2) contains flag bits to differentiate between a:

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

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

The PCON register bits are shown in Register 2-6.

REGISTER 2-6: PCON: POWER CONTROL REGISTER

U-0	U-0	U-0	U-0	U-0	U-0	R/W-0	R/W-0(1)
—	—	—	—	—	—	$\overline{POR}$	$\overline{BOR}$
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 **$\overline{POR}$:** 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)

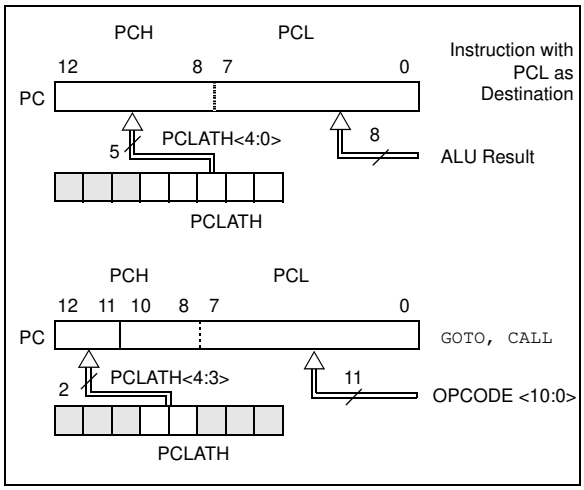
bit 0 **$\overline{BOR}$:** Brown-out Reset Status bit
1 = No Brown-out Reset occurred
0 = A Brown-out Reset occurred (must be set in software after a Brown-out Reset occurs)

Note 1: Reads as '0' if Brown-out Reset is disabled.

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 2-5 shows the two situations for the loading of the PC. The upper example in Figure 2-5 shows how the PC is loaded on a write to PCL (PCLATH<4:0> → PCH). The lower example in Figure 2-5 shows how the PC is loaded during a CALL or GOTO instruction (PCLATH<4:3> → PCH).

FIGURE 2-5: 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 PIC16F610/616/16HV610/616 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 a 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.

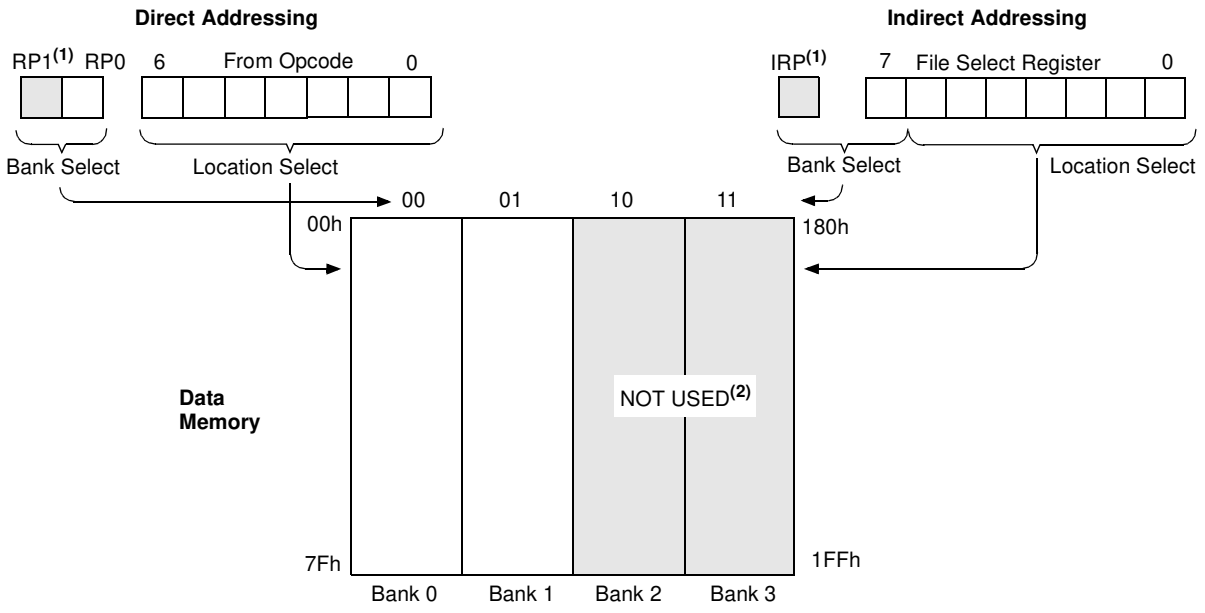
Indirect addressing is possible by using the INDF register. Any instruction using the INDF register actually accesses data pointed to by the File Select Register (FSR). Reading INDF itself indirectly will produce 00h. Writing to the INDF register indirectly results in a no operation (although Status 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 2-7.

A simple program to clear RAM location 40h-4Fh using indirect addressing is shown in Example 2-1.

EXAMPLE 2-1: INDIRECT ADDRESSING

```
        MOVLW   0x40    ;initialize pointer
        MOVWF   FSR     ;to RAM
NEXT    CLRF    INDF    ;clear INDF register
        INCF    FSR, F  ;inc pointer
        BTFSS   FSR,4   ;all done?
        GOTO    NEXT    ;no clear next
CONTINUE                ;yes continue
```

FIGURE 2-6: DIRECT/INDIRECT ADDRESSING PIC16F610/16HV610



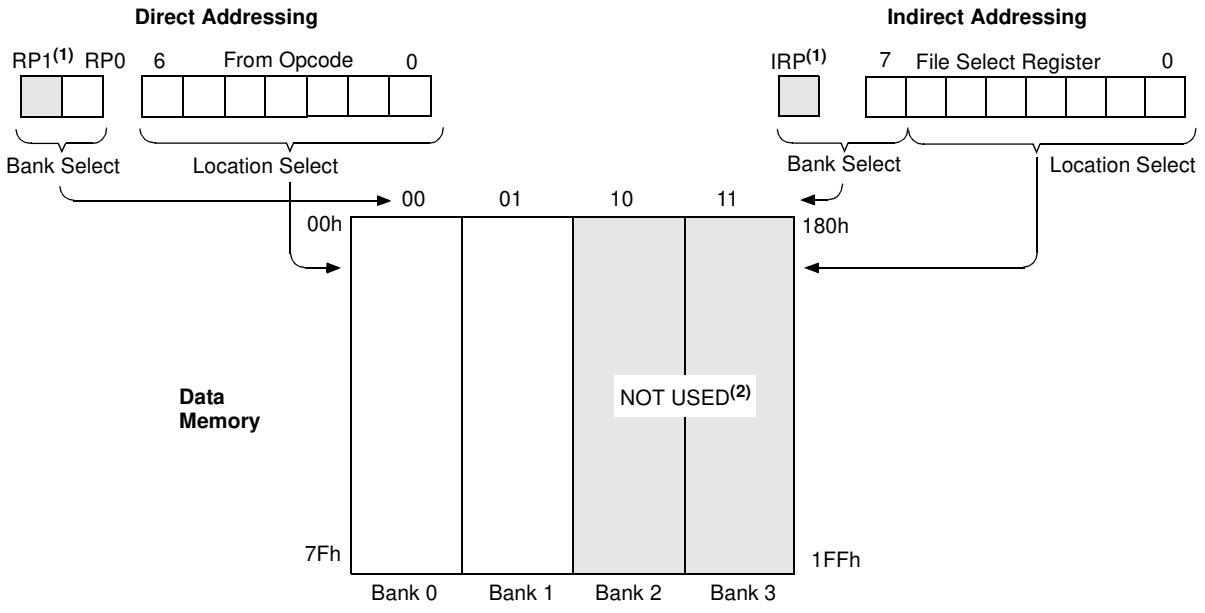
For memory map detail, see Figure 2-3.

Unimplemented data memory locations, read as '0'.

Note 1: The RP1 and IRP bits are reserved; always maintain these bits clear.

2: Accesses in Bank 2 and Bank 3 are mirrored back into Bank 0 and Bank 1, respectively.

FIGURE 2-7: DIRECT/INDIRECT ADDRESSING PIC16F616/16HV616



For memory map detail, see Figure 2-4.

Unimplemented data memory locations, read as '0'.

Note 1: The RP1 and IRP bits are reserved; always maintain these bits clear.

2: Accesses in Bank 2 and Bank 3 are mirrored back into Bank 0 and Bank 1, respectively.