



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

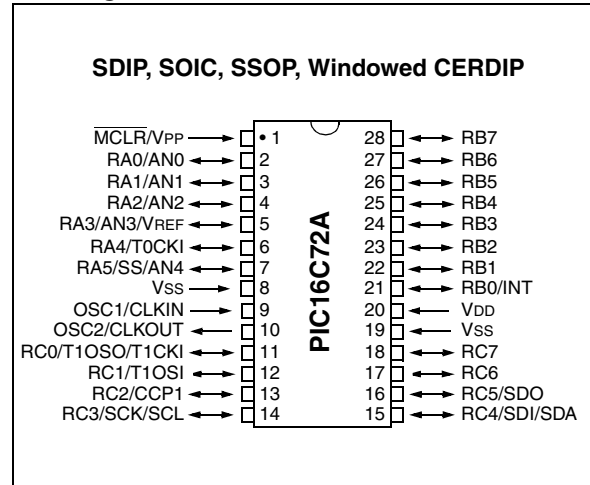


28-Pin 8-Bit CMOS Microcontrollers

Microcontroller Core Features:

- High-performance RISC CPU
- Only 35 single word instructions to learn
- All single cycle instructions except for program branches, which are two cycle
- Operating speed: DC - 20 MHz clock input
DC - 200 ns instruction cycle
- 2K x 14 words of Program Memory,
128 x 8 bytes of Data Memory (RAM)
- Interrupt capability
- Eight level deep hardware stack
- Direct, indirect, and relative addressing modes
- Power-on Reset (POR)
- Power-up Timer (PWRT) and
Oscillator Start-up Timer (OST)
- Watchdog Timer (WDT) with its own on-chip RC
oscillator for reliable operation
- Brown-out detection circuitry for
Brown-out Reset (BOR)
- Programmable code-protection
- Power saving SLEEP mode
- Selectable oscillator options
- Low-power, high-speed CMOS EPROM
technology
- Fully static design
- In-Circuit Serial Programming™ (ICSP)
- Wide operating voltage range: 2.5V to 5.5V
- High Sink/Source Current 25/25 mA
- Commercial, Industrial and Extended temperature
ranges
- Low-power consumption:
 - < 2 mA @ 5V, 4 MHz
 - 22.5 µA typical @ 3V, 32 kHz
 - < 1 µA typical standby current

Pin Diagram

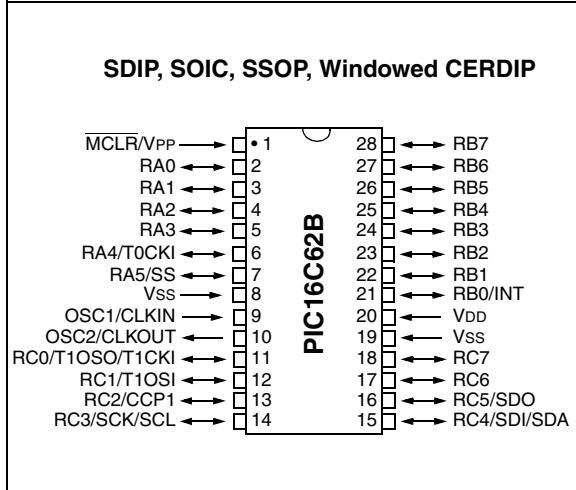


Peripheral Features:

- Timer0: 8-bit timer/counter with 8-bit prescaler
- Timer1: 16-bit timer/counter with prescaler,
can be incremented during sleep via external
crystal/clock
- Timer2: 8-bit timer/counter with 8-bit period
register, prescaler and postscaler
- Capture, Compare, PWM module
- Capture is 16-bit, max. resolution is 12.5 ns,
Compare is 16-bit, max. resolution is 200 ns,
PWM maximum resolution is 10-bit
- 8-bit multi-channel Analog-to-Digital converter
- Synchronous Serial Port (SSP) with Enhanced
SPI and I²C™

PIC16C62B/72A

Pin Diagrams



Key Features PIC® Mid-Range Reference Manual (DS33023)	PIC16C62B	PIC16C72A
Operating Frequency	DC - 20 MHz	DC - 20 MHz
Resets (and Delays)	POR, BOR (PWRT, OST)	POR, BOR (PWRT, OST)
Program Memory (14-bit words)	2K	2K
Data Memory (bytes)	128	128
Interrupts	7	8
I/O Ports	Ports A,B,C	Ports A,B,C
Timers	3	3
Capture/Compare/PWM modules	1	1
Serial Communications	SSP	SSP
8-bit Analog-to-Digital Module	—	5 input channels

Table of Contents

1.0 Device Overview	5
2.0 Memory Organization	7
3.0 I/O Ports	19
4.0 Timer0 Module	25
5.0 Timer1 Module	27
6.0 Timer2 Module	31
7.0 Capture/Compare/PWM (CCP) Module	33
8.0 Synchronous Serial Port (SSP) Module	39
9.0 Analog-to-Digital Converter (A/D) Module	49
10.0 Special Features of the CPU	55
11.0 Instruction Set Summary	67
12.0 Development Support	75
13.0 Electrical Characteristics	81
14.0 DC and AC Characteristics Graphs and Tables	103
15.0 Packaging Information	105
Appendix A: Revision History	111
Appendix B: Conversion Considerations	111
Appendix C: Migration from Base-line to Mid-Range Devices	112
Index	113
On-Line Support	117
Reader Response	118
PIC16C62B/72A Product Identification System	119

To Our Valued Customers

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.

New Customer Notification System

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

Errata

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

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

- Microchip's Worldwide Web site; <http://www.microchip.com>
- Your local Microchip sales office (see last page)
- The Microchip Corporate Literature Center; U.S. FAX: (480) 786-7277

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

Corrections to this Data Sheet

We constantly strive to improve the quality of all our products and documentation. We have spent a great deal of time to ensure that this document is correct. However, we realize that we may have missed a few things. If you find any information that is missing or appears in error, please:

- Fill out and mail in the reader response form in the back of this data sheet.
- E-mail us at webmaster@microchip.com.

We appreciate your assistance in making this a better document.

PIC16C62B/72A

NOTES:

1.0 DEVICE OVERVIEW

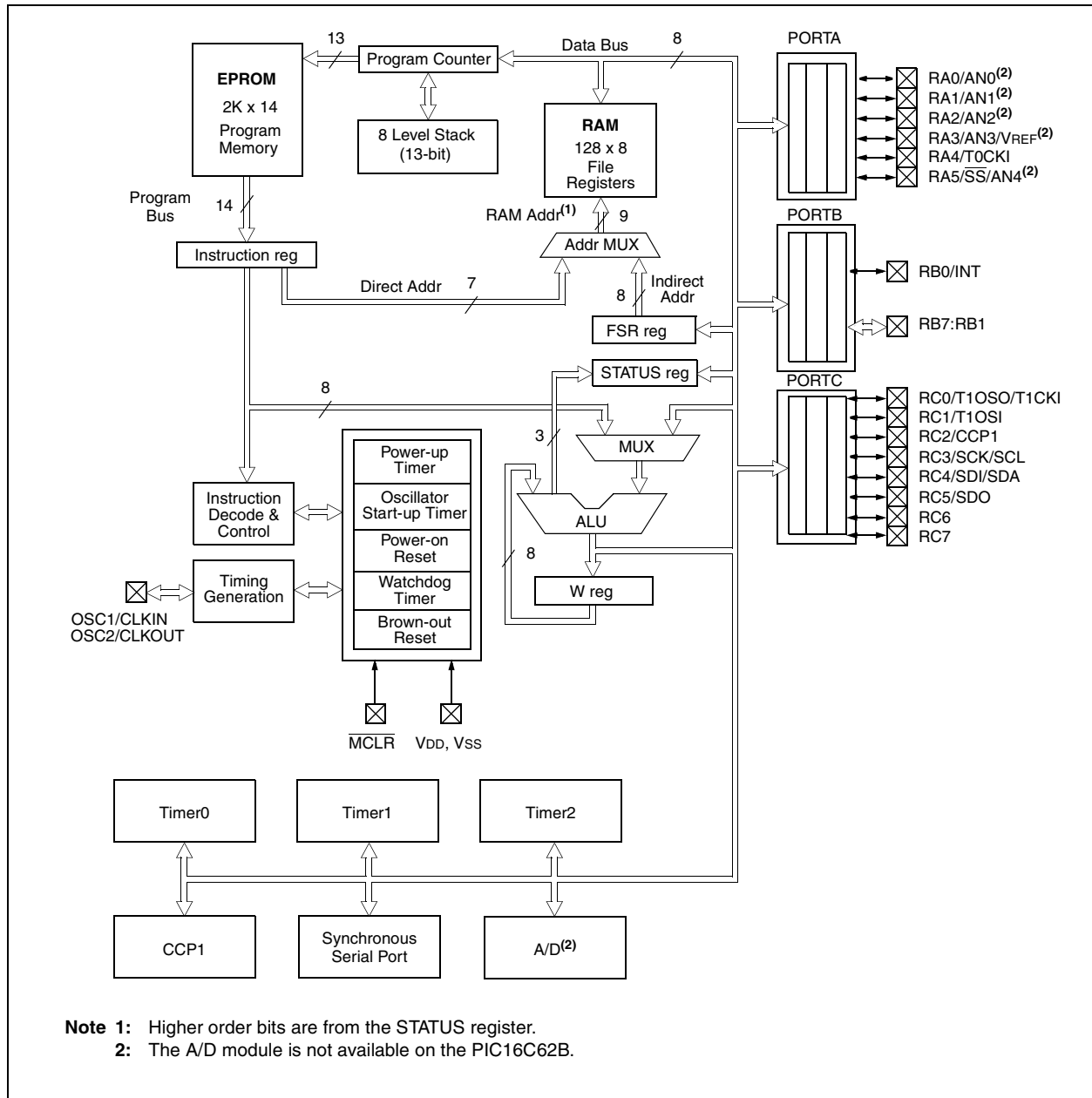
This document contains device-specific information. Additional information may be found in the PIC® MCU Mid-Range Reference Manual, (DS33023), which may be obtained from your local Microchip Sales Representative or downloaded from the Microchip website. The Reference Manual should be considered a complementary document to this data sheet, and is highly recommended reading for a better understanding of the device architecture and operation of the peripheral modules.

There are two devices (PIC16C62B, PIC16C72A) covered by this datasheet. The PIC16C62B does not have the A/D module implemented.

Figure 1-1 is the block diagram for both devices. The pinouts are listed in Table 1-1.

Figure 1-1 is the block diagram for both devices. The pinouts are listed in Table 1-1.

FIGURE 1-1: PIC16C62B/PIC16C72A BLOCK DIAGRAM



PIC16C62B/72A

TABLE 1-1 PIC16C62B/PIC16C72A PINOUT DESCRIPTION

Pin Name	DIP Pin#	SOIC Pin#	I/O/P Type	Buffer Type	Description
OSC1/CLKIN	9	9	I	ST/CMOS ⁽³⁾	Oscillator crystal input/external clock source input.
OSC2/CLKOUT	10	10	O	—	Oscillator crystal output. Connects to crystal or resonator in crystal oscillator mode. In RC mode, the OSC2 pin outputs CLKOUT which has 1/4 the frequency of OSC1, and denotes the instruction cycle rate.
MCLR/VPP	1	1	I/P	ST	Master clear (reset) input or programming voltage input. This pin is an active low reset to the device.
RA0/AN0 ⁽⁴⁾	2	2	I/O	TTL	<p>PORTA is a bi-directional I/O port.</p> <p>RA0 can also be analog input 0</p> <p>RA1 can also be analog input 1</p> <p>RA2 can also be analog input 2</p> <p>RA3 can also be analog input 3 or analog reference voltage</p> <p>RA4 can also be the clock input to the Timer0 module. Output is open drain type.</p> <p>RA5 can also be analog input 4 or the slave select for the synchronous serial port.</p>
RA1/AN1 ⁽⁴⁾	3	3	I/O	TTL	
RA2/AN2 ⁽⁴⁾	4	4	I/O	TTL	
RA3/AN3/VREF ⁽⁴⁾	5	5	I/O	TTL	
RA4/T0CKI	6	6	I/O	ST	
RA5/SS/AN4 ⁽⁴⁾	7	7	I/O	TTL	
RB0/INT	21	21	I/O	TTL/ST ⁽¹⁾	<p>PORTB is a bi-directional I/O port. PORTB can be software programmed for internal weak pull-up on all inputs.</p> <p>RB0 can also be the external interrupt pin.</p> <p>Interrupt on change pin.</p> <p>Interrupt on change pin.</p> <p>Interrupt on change pin. Serial programming clock.</p> <p>Interrupt on change pin. Serial programming data.</p>
RB1	22	22	I/O	TTL	
RB2	23	23	I/O	TTL	
RB3	24	24	I/O	TTL	
RB4	25	25	I/O	TTL	
RB5	26	26	I/O	TTL	
RB6	27	27	I/O	TTL/ST ⁽²⁾	
RB7	28	28	I/O	TTL/ST ⁽²⁾	
RC0/T1OSO/T1CKI	11	11	I/O	ST	<p>PORTC is a bi-directional I/O port.</p> <p>RC0 can also be the Timer1 oscillator output or Timer1 clock input.</p> <p>RC1 can also be the Timer1 oscillator input.</p> <p>RC2 can also be the Capture1 input/Compare1 output/PWM1 output.</p> <p>RC3 can also be the synchronous serial clock input/output for both SPI and I²C modes.</p> <p>RC4 can also be the SPI Data In (SPI mode) or data I/O (I²C mode).</p> <p>RC5 can also be the SPI Data Out (SPI mode).</p>
RC1/T1OSI	12	12	I/O	ST	
RC2/CCP1	13	13	I/O	ST	
RC3/SCK/SCL	14	14	I/O	ST	
RC4/SDI/SDA	15	15	I/O	ST	
RC5/SDO	16	16	I/O	ST	
RC6	17	17	I/O	ST	
RC7	18	18	I/O	ST	
Vss	8, 19	8, 19	P	—	Ground reference for logic and I/O pins.
VDD	20	20	P	—	Positive supply for logic and I/O pins.

Legend: I = input O = output I/O = input/output P = power or program
 — = Not used TTL = TTL input ST = Schmitt Trigger input

- Note 1:** This buffer is a Schmitt Trigger input when configured as the external interrupt.
Note 2: This buffer is a Schmitt Trigger input when used in serial programming mode.
Note 3: This buffer is a Schmitt Trigger input when configured in RC oscillator mode and a CMOS input otherwise.
Note 4: The A/D module is not available on the PIC16C62B.

2.0 MEMORY ORGANIZATION

There are two memory blocks in each of these micro-controllers. Each block (Program Memory and Data Memory) has its own bus, so that concurrent access can occur.

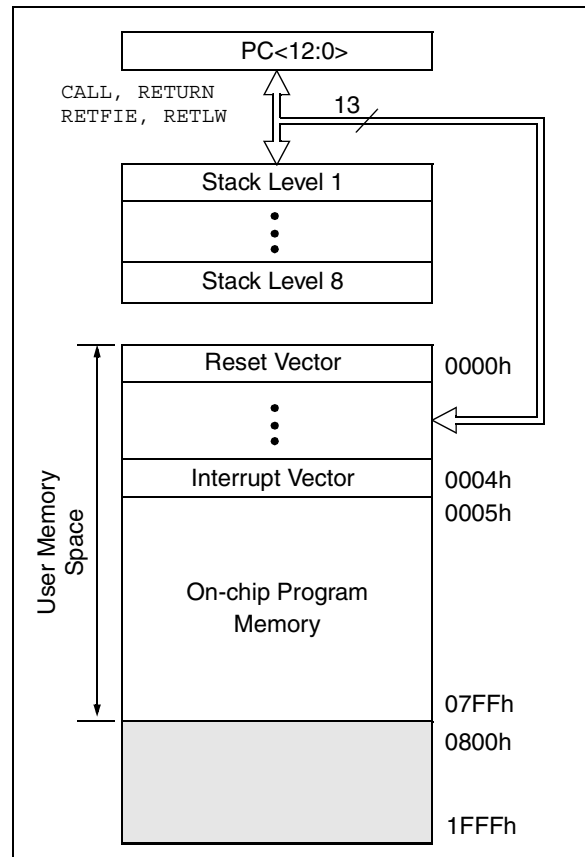
Additional information on device memory may be found in the PICmicro™ Mid-Range Reference Manual, (DS33023).

2.1 Program Memory Organization

The PIC16C62B/72A devices have a 13-bit program counter capable of addressing an 8K x 14 program memory space. Each device has 2K x 14 words of program memory. Accessing a location above 07FFh will cause a wraparound.

The reset vector is at 0000h and the interrupt vector is at 0004h.

FIGURE 2-1: PROGRAM MEMORY MAP AND STACK



PIC16C62B/72A

2.2 Data Memory Organization

The data memory is partitioned into multiple banks which contain the General Purpose Registers and the Special Function Registers. Bits RP1 and RP0 are the bank select bits.

RP1 ⁽¹⁾	RP0	(STATUS<6:5>)
--------------------	-----	---------------

= 00 → Bank0

= 01 → Bank1

= 10 → Bank2 (not implemented)

= 11 → Bank3 (not implemented)

Note 1: Maintain this bit clear to ensure upward compatibility with future products.

Each bank extends up to 7Fh (128 bytes). The lower locations of each bank are reserved for the Special Function Registers. Above the Special Function Registers are General Purpose Registers, implemented as static RAM. All implemented banks contain Special Function Registers. Some “high use” Special Function Registers from one bank may be mirrored in another bank for code reduction and quicker access.

2.2.1 GENERAL PURPOSE REGISTER FILE

The register file can be accessed either directly, or indirectly through the File Select Register FSR (Section 2.5).

FIGURE 2-2: REGISTER FILE MAP

File Address	Bank 0	Bank 1	File Address
00h	INDF ⁽¹⁾	INDF ⁽¹⁾	80h
01h	TMR0	OPTION_REG	81h
02h	PCL	PCL	82h
03h	STATUS	STATUS	83h
04h	FSR	FSR	84h
05h	PORTA	TRISA	85h
06h	PORTB	TRISB	86h
07h	PORTC	TRISC	87h
08h	—	—	88h
09h	—	—	89h
0Ah	PCLATH	PCLATH	8Ah
0Bh	INTCON	INTCON	8Bh
0Ch	PIR1	PIE1	8Ch
0Dh	—	—	8Dh
0Eh	TMR1L	PCON	8Eh
0Fh	TMR1H	—	8Fh
10h	T1CON	—	90h
11h	TMR2	—	91h
12h	T2CON	PR2	92h
13h	SSPBUF	SSPADD	93h
14h	SSPCON	SSPSTAT	94h
15h	CCPR1L	—	95h
16h	CCPR1H	—	96h
17h	CCP1CON	—	97h
18h	—	—	98h
19h	—	—	99h
1Ah	—	—	9Ah
1Bh	—	—	9Bh
1Ch	—	—	9Ch
1Dh	—	—	9Dh
1Eh	ADRES ⁽²⁾	—	9Eh
1Fh	ADCON0 ⁽²⁾	ADCON1 ⁽²⁾	9Fh
20h	General Purpose Registers	General Purpose Registers	A0h
			BFh
			C0h
7Fh			

Bank 0 Bank 1

Unimplemented data memory locations, read as '0'.

Note 1: Not a physical register.
Note 2: These registers are not implemented on the PIC16C62B, read as '0'.

2.2.2 SPECIAL FUNCTION REGISTERS

The Special Function Registers are registers used by the CPU and Peripheral Modules for controlling the desired operation of the device. These registers are implemented as static RAM. A list of these registers is given in Table 2-1.

The Special Function Registers can be classified into two sets; core (CPU) and peripheral. Those registers associated with the core functions are described in detail in this section. Those related to the operation of the peripheral features are described in detail in the peripheral feature section.

TABLE 2-1 SPECIAL FUNCTION REGISTER SUMMARY

Addr	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on: POR, BOR	Value on all other resets (4)		
Bank 0													
00h	INDF ⁽¹⁾	Addressing this location uses contents of FSR to address data memory (not a physical register)								0000 0000	0000 0000		
01h	TMR0	Timer0 module's register								xxxx xxxx	uuuu uuuu		
02h	PCL ⁽¹⁾	Program Counter's (PC) Least Significant Byte								0000 0000	0000 0000		
03h	STATUS ⁽¹⁾	IRP ⁽⁵⁾	RP1 ⁽⁵⁾	RP0	T \bar{O}	PD	Z	DC	C	0001 1xxx	000q quuu		
04h	FSR ⁽¹⁾	Indirect data memory address pointer								xxxx xxxx	uuuu uuuu		
05h	PORTA ^(6,7)	—	—	PORTA Data Latch when written: PORTA pins when read								--0x 0000	--0u 0000
06h	PORTB ^(6,7)	PORTB Data Latch when written: PORTB pins when read								xxxx xxxx	uuuu uuuu		
07h	PORTC ^(6,7)	PORTC Data Latch when written: PORTC pins when read								xxxx xxxx	uuuu uuuu		
08h-09h	—	Unimplemented								—	—		
0Ah	PCLATH ^(1,2)	—	—	—	Write Buffer for the upper 5 bits of the Program Counter						---0 0000	---0 0000	
0Bh	INTCON ⁽¹⁾	GIE	PEIE	T0IE	INTE	RBIE	T0IF	INTF	RBIF	0000 000x	0000 000u		
0Ch	PIR1	—	ADIF ⁽³⁾	—	—	SSPIF	CCP1IF	TMR2IF	TMR1IF	-0-- 0000	-0-- 0000		
0Dh	—	Unimplemented								—	—		
0Eh	TMR1L	Holding register for the Least Significant Byte of the 16-bit TMR1 register								xxxx xxxx	uuuu uuuu		
0Fh	TMR1H	Holding register for the Most Significant Byte of the 16-bit TMR1 register								xxxx xxxx	uuuu uuuu		
10h	T1CON	—	—	T1CKPS1	T1CKPS0	T1OSCEN	T1SYN \bar{C}	TMR1CS	TMR1ON	--00 0000	--uu uuuu		
11h	TMR2	Timer2 module's register								0000 0000	0000 0000		
12h	T2CON	—	TOUTPS3	TOUTPS2	TOUTPS1	TOUTPS0	TMR2ON	T2CKPS1	T2CKPS0	-000 0000	-000 0000		
13h	SSPBUF	Synchronous Serial Port Receive Buffer/Transmit Register								xxxx xxxx	uuuu uuuu		
14h	SSPCON	WCOL	SSPOV	SSPEN	CKP	SSPM3	SSPM2	SSPM1	SSPM0	0000 0000	0000 0000		
15h	CCPR1L	Capture/Compare/PWM Register1 (LSB)								xxxx xxxx	uuuu uuuu		
16h	CCPR1H	Capture/Compare/PWM Register1 (MSB)								xxxx xxxx	uuuu uuuu		
17h	CCP1CON	—	—	CCP1X	CCP1Y	CCP1M3	CCP1M2	CCP1M1	CCP1M0	--00 0000	--00 0000		
18h-1Dh	—	Unimplemented								—	—		
1Eh	ADRES ⁽³⁾	A/D Result Register								xxxx xxxx	uuuu uuuu		
1Fh	ADCON0 ⁽³⁾	ADCS1	ADCS0	CHS2	CHS1	CHS0	GO/DONE	—	ADON	0000 00-0	0000 00-0		

Legend: x = unknown, u = unchanged, q = value depends on condition, - = unimplemented, read as '0', Shaded locations are unimplemented, read as '0'.

- Note 1:** These registers can be addressed from either bank.
Note 2: The upper byte of the program counter is not directly accessible. PCLATH is a holding register for PC<12:8> whose contents are transferred to the upper byte of the program counter.
Note 3: A/D not implemented on the PIC16C62B, maintain as '0'.
Note 4: Other (non power-up) resets include: external reset through \overline{MCLR} and the Watchdog Timer Reset.
Note 5: The IRP and RP1 bits are reserved. Always maintain these bits clear.
Note 6: On any device reset, these pins are configured as inputs.
Note 7: This is the value that will be in the port output latch.

PIC16C62B/72A

TABLE 2-1 SPECIAL FUNCTION REGISTER SUMMARY (Cont'd)

Addr	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on: POR, BOR	Value on all other resets (4)
Bank 1											
80h	INDF ⁽¹⁾	Addressing this location uses contents of FSR to address data memory (not a physical register)								0000 0000	0000 0000
81h	OPTION_REG	RBP \bar{U}	INTEDG	T0CS	T0SE	PSA	PS2	PS1	PS0	1111 1111	1111 1111
82h	PCL ⁽¹⁾	Program Counter's (PC) Least Significant Byte								0000 0000	0000 0000
83h	STATUS ⁽¹⁾	IRP ⁽⁵⁾	RP1 ⁽⁵⁾	RP0	$\bar{T}O$	$\bar{P}D$	Z	DC	C	0001 1xxx	000q quuu
84h	FSR ⁽¹⁾	Indirect data memory address pointer								xxxx xxxx	uuuu uuuu
85h	TRISA	—	—	PORTA Data Direction Register						--11 1111	--11 1111
86h	TRISB	PORTB Data Direction Register								1111 1111	1111 1111
87h	TRISC	PORTC Data Direction Register								1111 1111	1111 1111
88h-89h	—	Unimplemented								—	—
8Ah	PCLATH ^(1,2)	—	—	—	Write Buffer for the upper 5 bits of the Program Counter					---0 0000	---0 0000
8Bh	INTCON ⁽¹⁾	GIE	PEIE	T0IE	INTE	RBIE	T0IF	INTF	RBIF	0000 000x	0000 000u
8Ch	PIE1	—	ADIE ⁽³⁾	—	—	SSPIE	CCP1IE	TMR2IE	TMR1IE	-0-- 0000	-0-- 0000
8Dh	—	Unimplemented								—	—
8Eh	PCON	—	—	—	—	—	—	POR	BOR	---- --qq	---- --uu
8Fh-91h	—	Unimplemented								—	—
92h	PR2	Timer2 Period Register								1111 1111	1111 1111
93h	SSPADD	Synchronous Serial Port (I ² C mode) Address Register								0000 0000	0000 0000
94h	SSPSTAT	SMP	CKE	D/ \bar{A}	P	S	R/ \bar{W}	UA	BF	0000 0000	0000 0000
95h-9Eh	—	Unimplemented								—	—
9Fh	ADCON1 ⁽³⁾	—	—	—	—	—	PCFG2	PCFG1	PCFG0	---- -000	---- -000

Legend: x = unknown, u = unchanged, q = value depends on condition, - = unimplemented, read as '0'.
Shaded locations are unimplemented, read as '0'.

Note 1: These registers can be addressed from either bank.

2: The upper byte of the program counter is not directly accessible. PCLATH is a holding register for PC<12:8> whose contents are transferred to the upper byte of the program counter.

3: A/D not implemented on the PIC16C62B, maintain as '0'.

4: Other (non power-up) resets include: external reset through \overline{MCLR} and the Watchdog Timer Reset.

5: The IRP and RP1 bits are reserved. Always maintain these bits clear.

6: On any device reset, these pins are configured as inputs.

7: This is the value that will be in the port output latch.

2.2.2.1 STATUS REGISTER

The STATUS register, shown in Register 2-1, contains the arithmetic status of the ALU, the RESET status and the bank select bits for data memory.

The STATUS register can be the destination for any instruction, as with any other register. If the STATUS register is the destination for an instruction that affects the Z, DC or C bits, the write to these three bits is disabled. These bits are set or cleared according to the device logic. The $\overline{\text{TO}}$ and $\overline{\text{PD}}$ bits are not writable. 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 the Z, C or DC bits from the STATUS register. For other instructions, not affecting any status bits, see the "Instruction Set Summary."

Note 1: The IRP and RP1 bits are reserved. Maintain these bits clear to ensure upward compatibility with future products.

Note 2: The C and DC bits operate as a borrow and digit borrow bit, respectively, in subtraction. See the `SUBLW` and `SUBWF` instructions.

REGISTER 2-1: STATUS REGISTER (ADDRESS 03h, 83h)

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	C
bit7							bit0
<p>bit 7: IRP: Register Bank Select bit (used for indirect addressing) (reserved, maintain clear)</p> <p>bit 6-5: RP1:RP0: Register Bank Select bits (used for direct addressing) 01 = Bank 1 (80h - FFh) 00 = Bank 0 (00h - 7Fh) Each bank is 128 bytes Note: RP1 is reserved, maintain clear</p> <p>bit 4: $\overline{\text{TO}}$: Time-out bit 1 = After power-up, <code>CLRWDT</code> instruction, or <code>SLEEP</code> instruction 0 = A WDT time-out occurred</p> <p>bit 3: $\overline{\text{PD}}$: Power-down bit 1 = After power-up or by the <code>CLRWDT</code> instruction 0 = By execution of the <code>SLEEP</code> instruction</p> <p>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</p> <p>bit 1: DC: Digit carry/borrow bit (<code>ADDWF</code>, <code>ADDLW</code>, <code>SUBLW</code>, <code>SUBWF</code> 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</p> <p>bit 0: C: Carry/borrow bit (<code>ADDWF</code>, <code>ADDLW</code>, <code>SUBLW</code>, <code>SUBWF</code> instructions) (for borrow, the polarity is reversed) 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</p> <p>Note: For borrow, the polarity is reversed. A subtraction is executed by adding the two's complement of the second operand. For rotate (<code>RRF</code>, <code>RLF</code>) instructions, this bit is loaded with either the high or low order bit of the source register.</p>							

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

PIC16C62B/72A

2.2.2.2 OPTION_REG REGISTER

The OPTION_REG register is a readable and writable register, which contains various control bits to configure the TMR0 prescaler/WDT postscaler (single assignable register known as the prescaler), the External INT Interrupt, TMR0 and the weak pull-ups on PORTB.

Note: To achieve a 1:1 prescaler assignment for the TMR0 register, assign the prescaler to the Watchdog Timer.

REGISTER 2-2: OPTION_REG REGISTER (ADDRESS 81h)

R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
RBP	INTEDG	T0CS	T0SE	PSA	PS2	PS1	PS0
							bit0
bit7							

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

bit 7: **RBP**: PORTB Pull-up Enable bit
1 = PORTB pull-ups are disabled
0 = PORTB pull-ups are enabled for all PORTB inputs

bit 6: **INTEDG**: Interrupt Edge Select bit
1 = Interrupt on rising edge of RB0/INT pin
0 = Interrupt on falling edge of RB0/INT pin

bit 5: **T0CS**: TMR0 Clock Source Select bit
1 = Transition on RA4/T0CKI pin
0 = Internal instruction cycle clock (CLKOUT)

bit 4: **T0SE**: TMR0 Source Edge Select bit
1 = Increment on high-to-low transition on RA4/T0CKI pin
0 = Increment on low-to-high transition on RA4/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: **PS2:PS0**: 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

2.2.2.3 INTCON REGISTER

The INTCON Register is a readable and writable register, which contains various interrupt enable and flag bits for the TMR0 register overflow, RB Port change and External RB0/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 (INTCON<7>). User software should ensure the appropriate interrupt flag bits are clear prior to enabling an interrupt.

REGISTER 2-3: INTCON REGISTER (ADDRESS 0Bh, 8Bh)

	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-x
bit7	GIE	PEIE	TOIE	INTE	RBIE	TOIF	INTF	RBIF
								bit0

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

bit 7: **GIE:** Global Interrupt Enable bit
1 = Enables all un-masked interrupts
0 = Disables all interrupts

bit 6: **PEIE:** Peripheral Interrupt Enable bit
1 = Enables all un-masked peripheral interrupts
0 = Disables all peripheral interrupts

bit 5: **TOIE:** TMR0 Overflow Interrupt Enable bit
1 = Enables the TMR0 interrupt
0 = Disables the TMR0 interrupt

bit 4: **INTE:** RB0/INT External Interrupt Enable bit
1 = Enables the RB0/INT external interrupt
0 = Disables the RB0/INT external interrupt

bit 3: **RBIE:** RB Port Change Interrupt Enable bit
1 = Enables the RB port change interrupt
0 = Disables the RB port change interrupt

bit 2: **TOIF:** TMR0 Overflow Interrupt Flag bit
1 = TMR0 register has overflowed (software must clear bit)
0 = TMR0 register did not overflow

bit 1: **INTF:** RB0/INT External Interrupt Flag bit
1 = The RB0/INT external interrupt occurred (software must clear bit)
0 = The RB0/INT external interrupt did not occur

bit 0: **RBIF:** RB Port Change Interrupt Flag bit
1 = At least one of the RB7:RB4 input pins have changed state (clear by reading PORTB)
0 = None of the RB7:RB4 input pins have changed state

PIC16C62B/72A

2.2.2.4 PIE1 REGISTER

This register contains the individual enable bits for the peripheral interrupts.

Note: Bit PEIE (INTCON<6>) must be set to enable any peripheral interrupt.

REGISTER 2-4: PIE1 REGISTER (ADDRESS 8Ch)

U-0	R/W-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0
—	ADIE ⁽¹⁾	—	—	SSPIE	CCP1IE	TMR2IE	TMR1IE

bit7 bit0

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

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

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

bit 5-4: **Unimplemented:** Read as '0'

bit 3: **SSPIE:** Synchronous Serial Port Interrupt Enable bit
1 = Enables the SSP interrupt
0 = Disables the SSP interrupt

bit 2: **CCP1IE:** CCP1 Interrupt Enable bit
1 = Enables the CCP1 interrupt
0 = Disables the CCP1 interrupt

bit 1: **TMR2IE:** TMR2 to PR2 Match Interrupt Enable bit
1 = Enables the TMR2 to PR2 match interrupt
0 = Disables the TMR2 to PR2 match interrupt

bit 0: **TMR1IE:** TMR1 Overflow Interrupt Enable bit
1 = Enables the TMR1 overflow interrupt
0 = Disables the TMR1 overflow interrupt

Note 1: The PIC16C62B does not have an A/D module. This bit location is reserved on these devices. Always maintain this bit clear.

2.2.2.5 PIR1 REGISTER

This register contains the individual flag bits for the Peripheral 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 (INTCON<7>). User software should ensure the appropriate interrupt flag bits are clear prior to enabling an interrupt.

REGISTER 2-5: PIR1 REGISTER (ADDRESS 0Ch)

	U-0	R/W-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	
bit7	—	ADIF ⁽¹⁾	—	—	SSPIF	CCP1IF	TMR2IF	TMR1IF	bit0
<div style="float: right; border: 1px solid black; padding: 5px; width: fit-content;"> R = Readable bit W = Writable bit U = Unimplemented bit, read as '0' - n = Value at POR reset </div> <p>bit 7: Unimplemented: Read as '0'</p> <p>bit 6: ADIF⁽¹⁾: A/D Converter Interrupt Flag bit 1 = An A/D conversion completed (must be cleared in software) 0 = The A/D conversion is not complete</p> <p>bit 5-4: Unimplemented: Read as '0'</p> <p>bit 3: SSPIF: Synchronous Serial Port Interrupt Flag bit 1 = The transmission/reception is complete (must be cleared in software) 0 = Waiting to transmit/receive</p> <p>bit 2: CCP1IF: CCP1 Interrupt Flag bit <u>Capture Mode</u> 1 = A TMR1 register capture occurred (must be cleared in software) 0 = No TMR1 register capture occurred <u>Compare Mode</u> 1 = A TMR1 register compare match occurred (must be cleared in software) 0 = No TMR1 register compare match occurred <u>PWM Mode</u> Unused in this mode</p> <p>bit 1: TMR2IF: TMR2 to PR2 Match Interrupt Flag bit 1 = TMR2 to PR2 match occurred (must be cleared in software) 0 = No TMR2 to PR2 match occurred</p> <p>bit 0: TMR1IF: TMR1 Overflow Interrupt Flag bit 1 = TMR1 register overflowed (must be cleared in software) 0 = TMR1 register did not overflow</p> <p>Note 1: The PIC16C62B does not have an A/D module. This bit location is reserved on these devices. Always maintain this bit clear.</p>									

PIC16C62B/72A

2.2.2.6 PCON REGISTER

The Power Control register (PCON) contains flag bits to allow differentiation between a Power-on Reset (POR), Brown-Out Reset (BOR) and resets from other sources. .

Note: On Power-on Reset, the state of the $\overline{\text{BOR}}$ bit is unknown and is not predictable. If the BODEN bit in the configuration word is set, the user must first set the BOR bit on a POR, and check it on subsequent resets. If BOR is cleared while POR remains set, a Brown-out reset has occurred. If the BODEN bit is clear, the BOR bit may be ignored.

REGISTER 2-6: PCON REGISTER (ADDRESS 8Eh)

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

bit7 bit0

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

bit 1: **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)

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

2.3 PCL and PCLATH

The program counter (PC) specifies the address of the instruction to fetch for execution. The PC is 13 bits wide. The low byte is called the PCL register and is readable and writable. The high byte is called the PCH register. This register contains the PC<12:8> bits and is not directly accessible. All updates to the PCH register go through the PCLATH register.

2.3.1 STACK

The stack allows any combination of up to 8 program calls and interrupts to occur. The stack contains the return address from this branch in program execution.

Mid-range devices have an 8 level deep hardware stack. The stack space is not part of either program or data space and the stack pointer is not accessible. 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 modified when the stack is PUSHed or POPed.

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).

2.4 Program Memory Paging

The `CALL` and `GOTO` instructions provide 11 bits of address to allow branching within any 2K program memory page. When doing a `CALL` or `GOTO` instruction, the upper bit of the address is provided by PCLATH<3>. The user must ensure that the page select bit is programmed to address the proper program memory page. If a return from a `CALL` instruction (or interrupt) is executed, the entire 13-bit PC is popped from the stack. Therefore, manipulation of the PCLATH<3> bit is not required for the return instructions.

PIC16C62B/72A

2.5 Indirect Addressing, INDF and FSR Registers

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

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

A simple program to clear RAM locations 20h-2Fh using indirect addressing is shown in Example 2-1.

EXAMPLE 2-1: HOW TO CLEAR RAM USING INDIRECT ADDRESSING

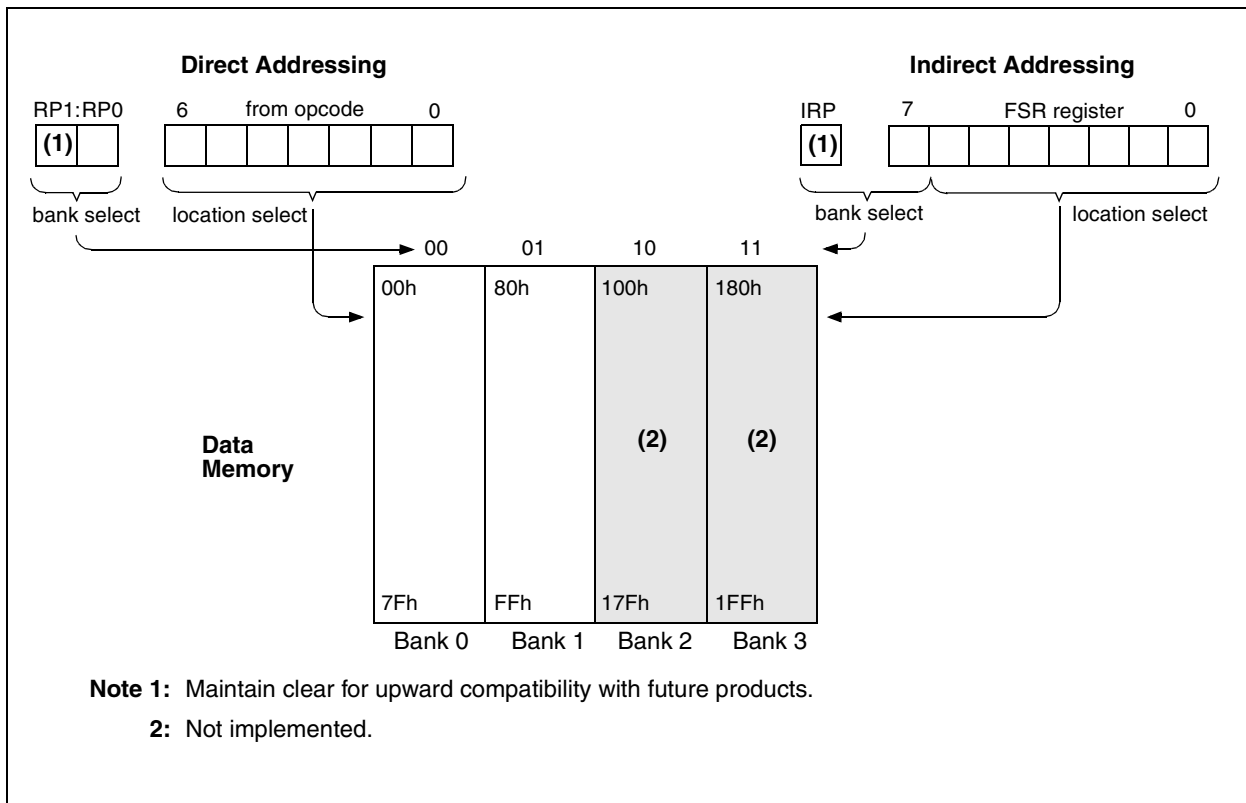
```

movlw 0x20 ;initialize pointer
movwf FSR ; to RAM
NEXT  clrf  INDF ;clear INDF register
      incf  FSR ;inc pointer
      btfss FSR,4 ;all done?
      goto NEXT ;NO, clear next

CONTINUE
      : ;YES, continue
    
```

An effective 9-bit address is obtained by concatenating the 8-bit FSR register and the IRP bit (STATUS<7>), as shown in Figure 2-3. However, IRP is not used in the PIC16C62B/72A.

FIGURE 2-3: DIRECT/INDIRECT ADDRESSING



3.0 I/O PORTS

Some I/O port pins are multiplexed with an alternate function for the peripheral features on the device. In general, when a peripheral is enabled, that pin may not be used as a general purpose I/O pin.

Additional information on I/O ports may be found in the PIC® MCU Mid-Range Reference Manual, (DS33023).

3.1 PORTA and the TRISA Register

PORTA is a 6-bit wide bi-directional port. The corresponding data direction register is TRISA. Setting a TRISA bit (=1) will make the corresponding PORTA pin an input, i.e., put the corresponding output driver in a hi-impedance mode. Clearing a TRISA bit (=0) will make the corresponding PORTA pin an output, (i.e., put the contents of the output latch on the selected pin).

The PORTA register reads the state 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.

Pin RA4 is multiplexed with the Timer0 module clock input to become the RA4/T0CKI pin. The RA4/T0CKI pin is a Schmitt Trigger input and an open drain output. All other RA port pins have TTL input levels and full CMOS output drivers.

Pin RA5 is multiplexed with the SSP to become the RA5/SS pin.

On the PIC16C72A device, other PORTA pins are multiplexed with analog inputs and analog VREF input. The operation of each pin is selected by clearing/setting the control bits in the ADCON1 register (A/D Control Register1).

Note: On a Power Reset, pins with analog functions are configured as analog inputs with digital input buffers disabled. A digital read of these pins will return '0'.

The TRISA register controls the direction of the RA 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.

FIGURE 3-1: BLOCK DIAGRAM OF RA3:RA0 AND RA5 PINS

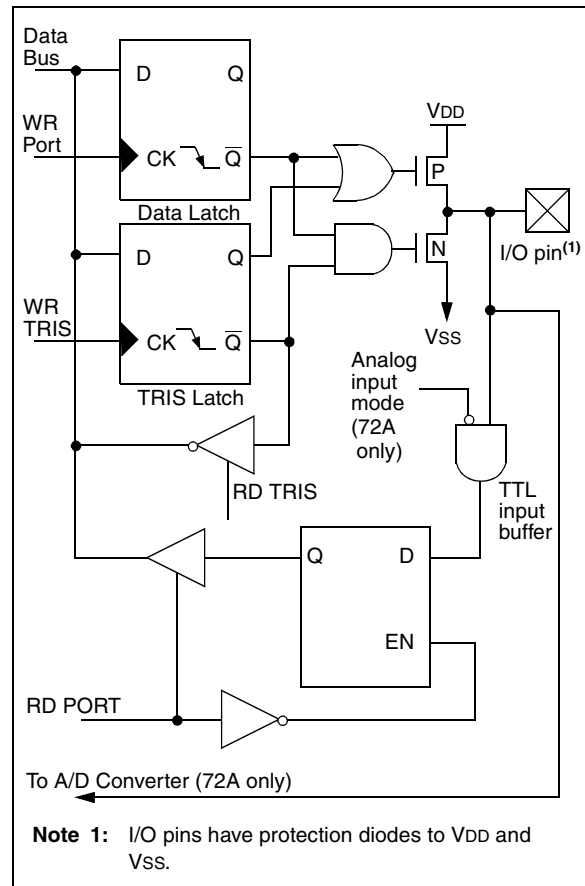
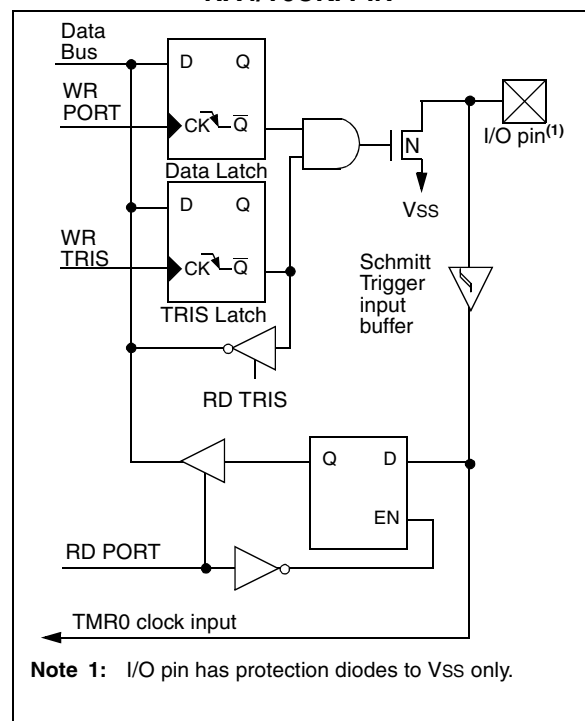


FIGURE 3-2: BLOCK DIAGRAM OF RA4/T0CKI PIN



PIC16C62B/72A

TABLE 3-1 PORTA FUNCTIONS

Name	Bit#	Buffer	Function
RA0/AN0	bit0	TTL	Input/output or analog input ⁽¹⁾
RA1/AN1	bit1	TTL	Input/output or analog input ⁽¹⁾
RA2/AN2	bit2	TTL	Input/output or analog input ⁽¹⁾
RA3/AN3/VREF	bit3	TTL	Input/output or analog input ⁽¹⁾ or VREF ⁽¹⁾
RA4/T0CKI	bit4	ST	Input/output or external clock input for Timer0 Output is open drain type
RA5/ \overline{SS} /AN4	bit5	TTL	Input/output or slave select input for synchronous serial port or analog input ⁽¹⁾

Legend: TTL = TTL input, ST = Schmitt Trigger input

Note 1: The PIC16C62B does not implement the A/D module.

TABLE 3-2 SUMMARY OF REGISTERS ASSOCIATED WITH PORTA

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on POR, BOR	Value on all other resets
05h	PORTA (for PIC16C72A only)	—	—	RA5	RA4	RA3	RA2	RA1	RA0	--0x 0000	--0u 0000
05h	PORTA (for PIC16C62B only)	—	—	RA5	RA4	RA3	RA2	RA1	RA0	--xx xxxx	--uu uuuu
85h	TRISA	—	—	PORTA Data Direction Register						--11 1111	--11 1111
9Fh	ADCON1 ⁽¹⁾	—	—	—	—	—	PCFG2	PCFG1	PCFG0	---- -000	---- -000

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

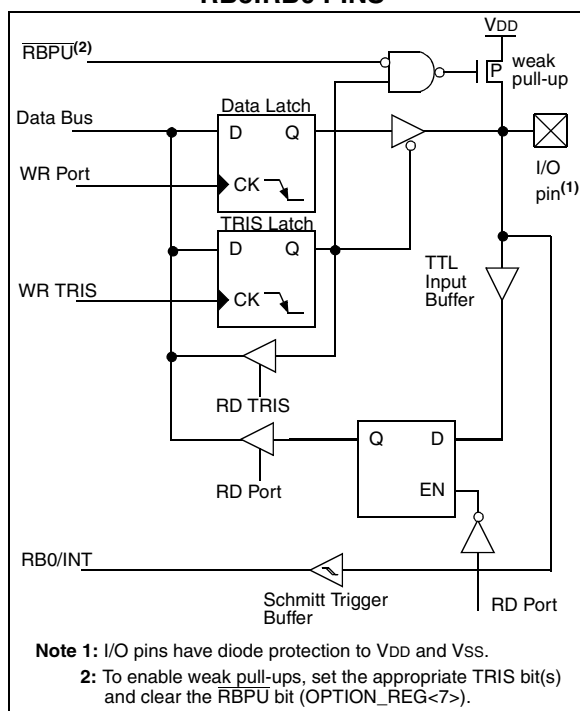
Note 1: The PIC16C62B does not implement the A/D module. Maintain this register clear.

3.2 PORTB and the TRISB Register

PORTB is an 8-bit wide bi-directional port. The corresponding data direction register is TRISB. Setting a TRISB bit (=1) will make the corresponding PORTB pin an input, (i.e., put the corresponding output driver in a hi-impedance mode). Clearing a TRISB bit (=0) will make the corresponding PORTB pin an output, (i.e., put the contents of the output latch on the selected pin).

Each of the PORTB pins has a weak internal pull-up. A single control bit can turn on all the pull-ups. This is performed by clearing bit \overline{RBPU} (OPTION_REG<7>). The 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.

FIGURE 3-3: BLOCK DIAGRAM OF RB3:RB0 PINS



Four of PORTB's pins, RB7:RB4, have an interrupt on change feature. Only pins configured as inputs can cause this interrupt to occur (i.e. any RB7:RB4 pin configured as an output is excluded from the interrupt on change comparison). The input pins (of RB7:RB4) are compared with the old value latched on the last read of PORTB. The "mismatch" outputs of RB7:RB4 are OR'ed together to generate the RB Port Change Interrupt with flag bit RBIF (INTCON<0>).

This interrupt can wake the device from SLEEP. The user, in the interrupt service routine, can clear the interrupt in the following manner:

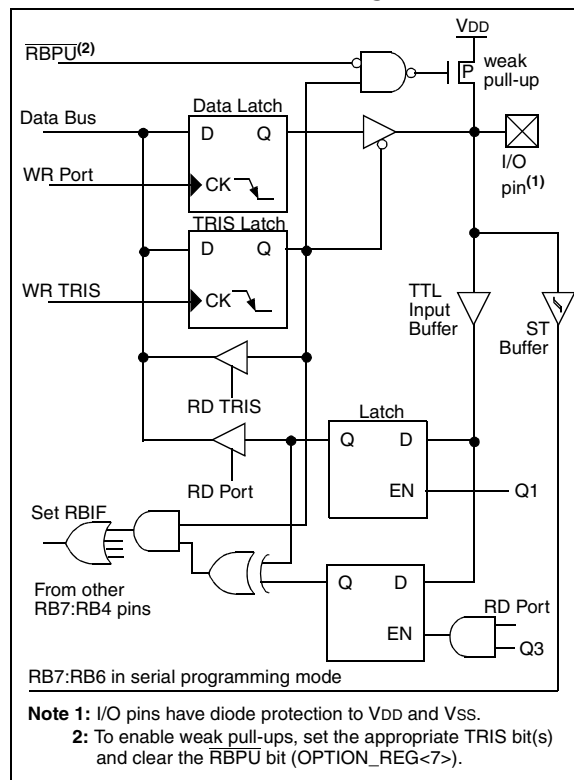
- Any read or write of PORTB. This will end the mismatch condition.
- Clear flag bit RBIF.

A mismatch condition will continue to set flag bit RBIF. Reading PORTB will end the mismatch condition and allow flag bit RBIF to be cleared.

The interrupt on change feature is recommended for wake-up on key depression operation and operations where PORTB is only used for the interrupt on change feature. Polling of PORTB is not recommended while using the interrupt on change feature.

RB0/INT is an external interrupt pin and is configured using the INTEDG bit (OPTION_REG<6>). RB0/INT is discussed in detail in Section 10.10.1.

FIGURE 3-4: BLOCK DIAGRAM OF RB7:RB4 PINS



PIC16C62B/72A

TABLE 3-3 PORTB FUNCTIONS

Name	Bit#	Buffer	Function
RB0/INT	bit0	TTL/ST ⁽¹⁾	Input/output pin or external interrupt input. Internal software programmable weak pull-up.
RB1	bit1	TTL	Input/output pin. Internal software programmable weak pull-up.
RB2	bit2	TTL	Input/output pin. Internal software programmable weak pull-up.
RB3	bit3	TTL	Input/output pin. Internal software programmable weak pull-up.
RB4	bit4	TTL	Input/output pin (with interrupt on change). Internal software programmable weak pull-up.
RB5	bit5	TTL	Input/output pin (with interrupt on change). Internal software programmable weak pull-up.
RB6	bit6	TTL/ST ⁽²⁾	Input/output pin (with interrupt on change). Internal software programmable weak pull-up. Serial programming clock.
RB7	bit7	TTL/ST ⁽²⁾	Input/output pin (with interrupt on change). Internal software programmable weak pull-up. Serial programming data.

Legend: TTL = TTL input, ST = Schmitt Trigger input

Note 1: This buffer is a Schmitt Trigger input when configured as the external interrupt.

2: This buffer is a Schmitt Trigger input when used in serial programming mode.

TABLE 3-4 SUMMARY OF REGISTERS ASSOCIATED WITH PORTB

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on: POR, BOR	Value on all other resets
06h	PORTB	RB7	RB6	RB5	RB4	RB3	RB2	RB1	RB0	xxxx xxxx	uuuu uuuu
86h	TRISB	PORTB Data Direction Register								1111 1111	1111 1111
81h	OPTION_REG	RBPU	INTEDG	T0CS	T0SE	PSA	PS2	PS1	PS0	1111 1111	1111 1111

Legend: x = unknown, u = unchanged. Shaded cells are not used by PORTB.

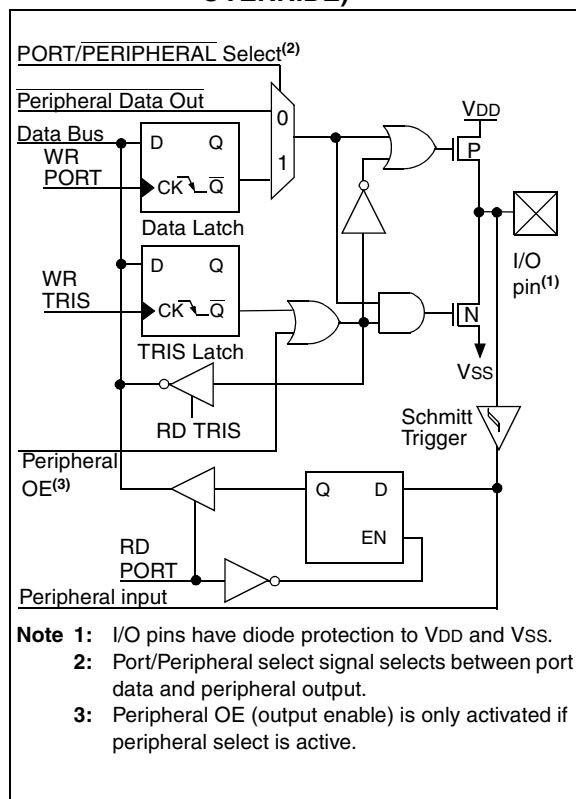
3.3 PORTC and the TRISC Register

PORTC is an 8-bit wide bi-directional port. The corresponding data direction register is TRISC. Setting a TRISC bit (=1) will make the corresponding PORTC pin an input, (i.e., put the corresponding output driver in a hi-impedance mode). Clearing a TRISC bit (=0) will make the corresponding PORTC pin an output, (i.e., put the contents of the output latch on the selected pin).

PORTC is multiplexed with several peripheral functions (Table 3-5). PORTC pins have Schmitt Trigger input buffers.

When enabling peripheral functions, care should be taken in defining TRIS bits for each PORTC pin. Some peripherals override the TRIS bit to make a pin an output, while other peripherals override the TRIS bit to make a pin an input. Since the TRIS bit override maybe in effect while the peripheral is enabled, read-modify-write instructions (*BSF*, *BCF*, *XORWF*) with TRISC as destination should be avoided. The user should refer to the corresponding peripheral section for the correct TRIS bit settings.

FIGURE 3-5: PORTC BLOCK DIAGRAM (PERIPHERAL OUTPUT OVERRIDE)



PIC16C62B/72A

TABLE 3-5 PORTC FUNCTIONS

Name	Bit#	Buffer Type	Function	TRISC Override
RC0/T1OSO/T1CKI	bit0	ST	Input/output port pin or Timer1 oscillator output/Timer1 clock input	Yes
RC1/T1OSI	bit1	ST	Input/output port pin or Timer1 oscillator input	Yes
RC2/CCP1	bit2	ST	Input/output port pin or Capture1 input/Compare1 output/PWM1 output	No
RC3/SCK/SCL	bit3	ST	RC3 can also be the synchronous serial clock for both SPI and I ² C modes.	No
RC4/SDI/SDA	bit4	ST	RC4 can also be the SPI Data In (SPI mode) or data I/O (I ² C mode).	No
RC5/SDO	bit5	ST	Input/output port pin or Synchronous Serial Port data output	No
RC6	bit6	ST	Input/output port pin	No
RC7	bit7	ST	Input/output port pin	No

Legend: ST = Schmitt Trigger input

TABLE 3-6 SUMMARY OF REGISTERS ASSOCIATED WITH PORTC

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on: POR, BOR	Value on all other resets
07h	PORTC	RC7	RC6	RC5	RC4	RC3	RC2	RC1	RC0	xxxx xxxx	uuuu uuuu
87h	TRISC	PORTC Data Direction Register								1111 1111	1111 1111

Legend: x = unknown, u = unchanged.

4.0 TIMER0 MODULE

The Timer0 module timer/counter has the following features:

- 8-bit timer/counter
 - Read and write
 - INT on overflow
- 8-bit software programmable prescaler
- INT or EXT clock select
 - EXT clock edge select

Figure 4-1 is a simplified block diagram of the Timer0 module.

Additional information on timer modules is available in the PIC® MCU Mid-Range Reference Manual, (DS33023).

4.1 Timer0 Operation

Timer0 can operate as a timer or as a counter.

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

Counter mode is selected by setting bit T0CS (OPTION_REG<5>). In counter mode, Timer0 will increment either on every rising or falling edge of pin RA4/T0CKI. The incrementing edge is determined by the Timer0 Source Edge Select bit T0SE (OPTION_REG<4>). Clearing bit T0SE selects the rising edge. Restrictions on the external clock input are discussed below.

When an external clock input is used for Timer0, it must meet certain requirements. The requirements ensure the external clock can be synchronized with the internal phase clock (Tosc). Also, there is a delay in the actual incrementing of Timer0 after synchronization.

Additional information on external clock requirements is available in the Electrical Specifications section of this manual, and in the PIC® MCU Mid-Range Reference Manual, (DS33023).

4.2 Prescaler

An 8-bit counter is available as a prescaler for the Timer0 module, or as a postscaler for the Watchdog Timer, respectively (Figure 4-2). For simplicity, this counter is being referred to as “prescaler” throughout this data sheet. There is only one prescaler available which is shared between the Timer0 module and the Watchdog Timer. A prescaler assignment for the Timer0 module means that there is no prescaler for the Watchdog Timer, and vice-versa.

The prescaler is not readable or writable.

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

Clearing bit PSA will assign the prescaler to the Timer0 module. When the prescaler is assigned to the Timer0 module, prescale values of 1:2, 1:4, ..., 1:256 are selectable.

Setting bit PSA will assign the prescaler to the Watchdog Timer (WDT). When the prescaler is assigned to the WDT, prescale values of 1:1, 1:2, ..., 1:128 are selectable.

When assigned to the Timer0 module, all instructions writing to the TMR0 register (e.g. CLRF 1, MOVWF 1, BSF 1, x....etc.) will clear the prescaler. When assigned to WDT, a CLRWDT instruction will clear the prescaler along with the WDT.

Note: Writing to TMR0 when the prescaler is assigned to Timer0 will clear the prescaler count, but will not change the prescaler assignment or ratio.

FIGURE 4-1: TIMER0 BLOCK DIAGRAM

