



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



OTP 8-Bit CMOS MCU with EEPROM Data Memory

Devices included in this data sheet:

- PIC16CE623
- PIC16CE624
- PIC16CE625

High Performance RISC CPU:

- Only 35 instructions to learn
- All single-cycle instructions (200 ns), except for program branches which are two-cycle
- Operating speed:
 - DC - 20 MHz clock input
 - DC - 200 ns instruction cycle

Device	Program Memory	RAM Data Memory	EEPROM Data Memory
PIC16CE623	512x14	96x8	128x8
PIC16CE624	1Kx14	96x8	128x8
PIC16CE625	2Kx14	128x8	128x8

- Interrupt capability
- 16 special function hardware registers
- 8-level deep hardware stack
- Direct, Indirect and Relative addressing modes

Peripheral Features:

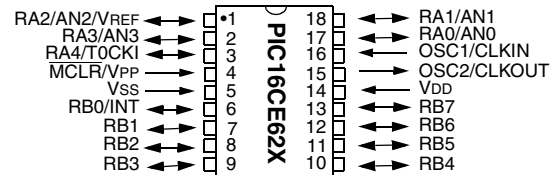
- 13 I/O pins with individual direction control
- High current sink/source for direct LED drive
- Analog comparator module with:
 - Two analog comparators
 - Programmable on-chip voltage reference (VREF) module
 - Programmable input multiplexing from device inputs and internal voltage reference
 - Comparator outputs can be output signals
- Timer0: 8-bit timer/counter with 8-bit programmable prescaler

Special Microcontroller Features:

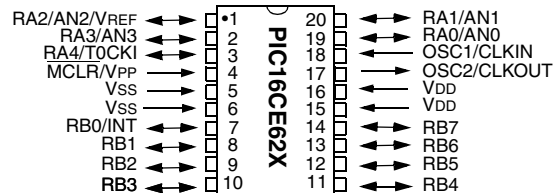
- In-Circuit Serial Programming (ICSP™) (via two pins)
- Power-on Reset (POR)
- Power-up Timer (PWRT) and Oscillator Start-up Timer (OST)
- Brown-out Reset
- Watchdog Timer (WDT) with its own on-chip RC oscillator for reliable operation

Pin Diagrams

PDIP, SOIC, Windowed CERDIP



SSOP



Special Microcontroller Features (cont'd)

- 1,000,000 erase/write cycle EEPROM data memory
- EEPROM data retention > 40 years
- Programmable code protection
- Power saving SLEEP mode
- Selectable oscillator options
- Four user programmable ID locations

CMOS Technology:

- Low-power, high-speed CMOS EPROM/EEPROM technology
- Fully static design
- Wide operating voltage range
 - 2.5V to 5.5V
- Commercial, industrial and extended temperature range
- Low power consumption
 - < 2.0 mA @ 5.0V, 4.0 MHz
 - 15 µA typical @ 3.0V, 32 kHz
 - < 1.0 µA typical standby current @ 3.0V

PIC16CE62X

Table of Contents

1.0	General Description	3
2.0	PIC16CE62X Device Varieties	5
3.0	Architectural Overview	7
4.0	Memory Organization	11
5.0	I/O Ports.....	23
6.0	EEPROM Peripheral Operation	29
7.0	Timer0 Module.....	35
8.0	Comparator Module	41
9.0	Voltage Reference Module	47
10.0	Special Features of the CPU	49
11.0	Instruction Set Summary	65
12.0	Development Support.....	77
13.0	Electrical Specifications	83
14.0	Packaging Information.....	97
	Appendix A: Code for Accessing EEPROM Data Memory	103
	Index	105
	On Line Support.....	107
	Reader Response	108
	PIC16CE62X Product Identification System	109

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.

1.0 GENERAL DESCRIPTION

The PIC16CE62X are 18 and 20-Pin EPROM-based members of the versatile PIC[®] family of low-cost, high-performance, CMOS, fully-static, 8-bit microcontrollers with EEPROM data memory.

All PIC[®] microcontrollers employ an advanced RISC architecture. The PIC16CE62X family has enhanced core features, eight-level deep stack, and multiple internal and external interrupt sources. The separate instruction and data buses of the Harvard architecture allow a 14-bit wide instruction word with separate 8-bit wide data. The two-stage instruction pipeline allows all instructions to execute in a single-cycle, except for program branches (which require two cycles). A total of 35 instructions (reduced instruction set) are available. Additionally, a large register set gives some of the architectural innovations used to achieve a very high performance.

PIC16CE62X microcontrollers typically achieve a 2:1 code compression and a 4:1 speed improvement over other 8-bit microcontrollers in their class.

The PIC16CE623 and PIC16CE624 have 96 bytes of RAM. The PIC16CE625 has 128 bytes of RAM. Each microcontroller contains a 128x8 EEPROM memory array for storing non-volatile information, such as calibration data or security codes. This memory has an endurance of 1,000,000 erase/write cycles and a retention of 40 plus years.

Each device has 13 I/O pins and an 8-bit timer/counter with an 8-bit programmable prescaler. In addition, the PIC16CE62X adds two analog comparators with a programmable on-chip voltage reference module. The comparator module is ideally suited for applications requiring a low-cost analog interface (e.g., battery chargers, threshold detectors, white goods controllers, etc).

PIC16CE62X devices have special features to reduce external components, thus reducing system cost, enhancing system reliability and reducing power consumption. There are four oscillator options, of which the single pin RC oscillator provides a low-cost solution, the LP oscillator minimizes power consumption, XT is a standard crystal, and the HS is for High Speed crystals. The SLEEP (power-down) mode offers power savings. The user can wake-up the chip from SLEEP through several external and internal interrupts and reset.

A highly reliable Watchdog Timer with its own on-chip RC oscillator provides protection against software lock-up.

A UV-erasable CERDIP-packaged version is ideal for code development, while the cost-effective One-Time Programmable (OTP) version is suitable for production in any volume.

Table 1-1 shows the features of the PIC16CE62X mid-range microcontroller families.

A simplified block diagram of the PIC16CE62X is shown in Figure 3-1.

The PIC16CE62X series fits perfectly in applications ranging from multi-pocket battery chargers to low-power remote sensors. The EPROM technology makes customization of application programs (detection levels, pulse generation, timers, etc.) extremely fast and convenient. The small footprint packages make this microcontroller series perfect for all applications with space limitations. Low-cost, low-power, high-performance, ease of use and I/O flexibility make the PIC16CE62X very versatile.

1.1 Development Support

The PIC16CE62X family is supported by a full-featured macro assembler, a software simulator, an in-circuit emulator, a low-cost development programmer and a full-featured programmer. A "C" compiler is also available.

PIC16CE62X

TABLE 1-1: PIC16CE62X FAMILY OF DEVICES

		PIC16CE623	PIC16CE624	PIC16CE625
Clock	Maximum Frequency of Operation (MHz)	20	20	20
Memory	EPROM Program Memory (x14 words)	512	1K	2K
	Data Memory (bytes)	96	96	128
Peripherals	EEPROM Data Memory (bytes)	128	128	128
	Timer Module(s)	TMR0	TMR0	TMR0
	Comparators(s)	2	2	2
	Internal Reference Voltage	Yes	Yes	Yes
Features	Interrupt Sources	4	4	4
	I/O Pins	13	13	13
	Voltage Range (Volts)	2.5-5.5	2.5-5.5	2.5-5.5
	Brown-out Reset	Yes	Yes	Yes
	Packages	18-pin DIP, SOIC; 20-pin SSOP	18-pin DIP, SOIC; 20-pin SSOP	18-pin DIP, SOIC; 20-pin SSOP

All PIC® Family devices have Power-on Reset, selectable Watchdog Timer, selectable code protect and high I/O current capability.
All PIC16CE62X Family devices use serial programming with clock pin RB6 and data pin RB7.

2.0 PIC16CE62X DEVICE VARIETIES

A variety of frequency ranges and packaging options are available. Depending on application and production requirements the proper device option can be selected using the information in the PIC16CE62X Product Identification System section at the end of this data sheet. When placing orders, please use this page of the data sheet to specify the correct part number.

2.1 UV Erasable Devices

The UV erasable version, offered in the CERDIP package is optimal for prototype development and pilot programs. This version can be erased and reprogrammed to any of the oscillator modes.

Microchip's PICSTART[®] and PRO MATE[®] programmers both support programming of the PIC16CE62X.

2.2 One-Time-Programmable (OTP) Devices

The availability of OTP devices is especially useful for customers who need the flexibility for frequent code updates and small volume applications. In addition to the program memory, the configuration bits must also be programmed.

2.3 Quick-Turn-Programming (QTP) Devices

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

2.4 Serialized Quick-Turn-Programming (SQTPSM) Devices

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

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

PIC16CE62X

NOTES:

3.0 ARCHITECTURAL OVERVIEW

The high performance of the PIC16CE62X family can be attributed to a number of architectural features commonly found in RISC microprocessors. To begin with, the PIC16CE62X uses a Harvard architecture in which program and data are accessed from separate memories using separate buses. This improves bandwidth over traditional von Neumann architecture where program and data are fetched from the same memory. Separating program and data memory further allows instructions to be sized differently than 8-bit wide data word. Instruction opcodes are 14-bits wide making it possible to have all single word instructions. A 14-bit wide program memory access bus fetches a 14-bit instruction in a single cycle. A two-stage pipeline overlaps fetch and execution of instructions. Consequently, all instructions (35) execute in a single-cycle (200 ns @ 20 MHz) except for program branches.

The table below lists program memory (EPROM), data memory (RAM) and non-volatile memory (EEPROM) for each PIC16CE62X device.

Device	Program Memory	RAM Data Memory	EEPROM Data Memory
PIC16CE623	512x14	96x8	128x8
PIC16CE624	1Kx14	96x8	128x8
PIC16CE625	2Kx14	128x8	128x8

The PIC16CE62X can directly or indirectly address its register files or data memory. All special function registers including the program counter are mapped in the data memory. The PIC16CE62X family has an orthogonal (symmetrical) instruction set that makes it possible to carry out any operation on any register using any addressing mode. This symmetrical nature and lack of 'special optimal situations' make programming with the PIC16CE62X simple yet efficient. In addition, the learning curve is reduced significantly.

The PIC16CE62X devices contain an 8-bit ALU and working register. The ALU is a general purpose arithmetic unit. It performs arithmetic and Boolean functions between data in the working register and any register file.

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

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

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

A simplified block diagram is shown in Figure 3-1, with a description of the device pins in Table 3-1.

PIC16CE62X

FIGURE 3-1: BLOCK DIAGRAM

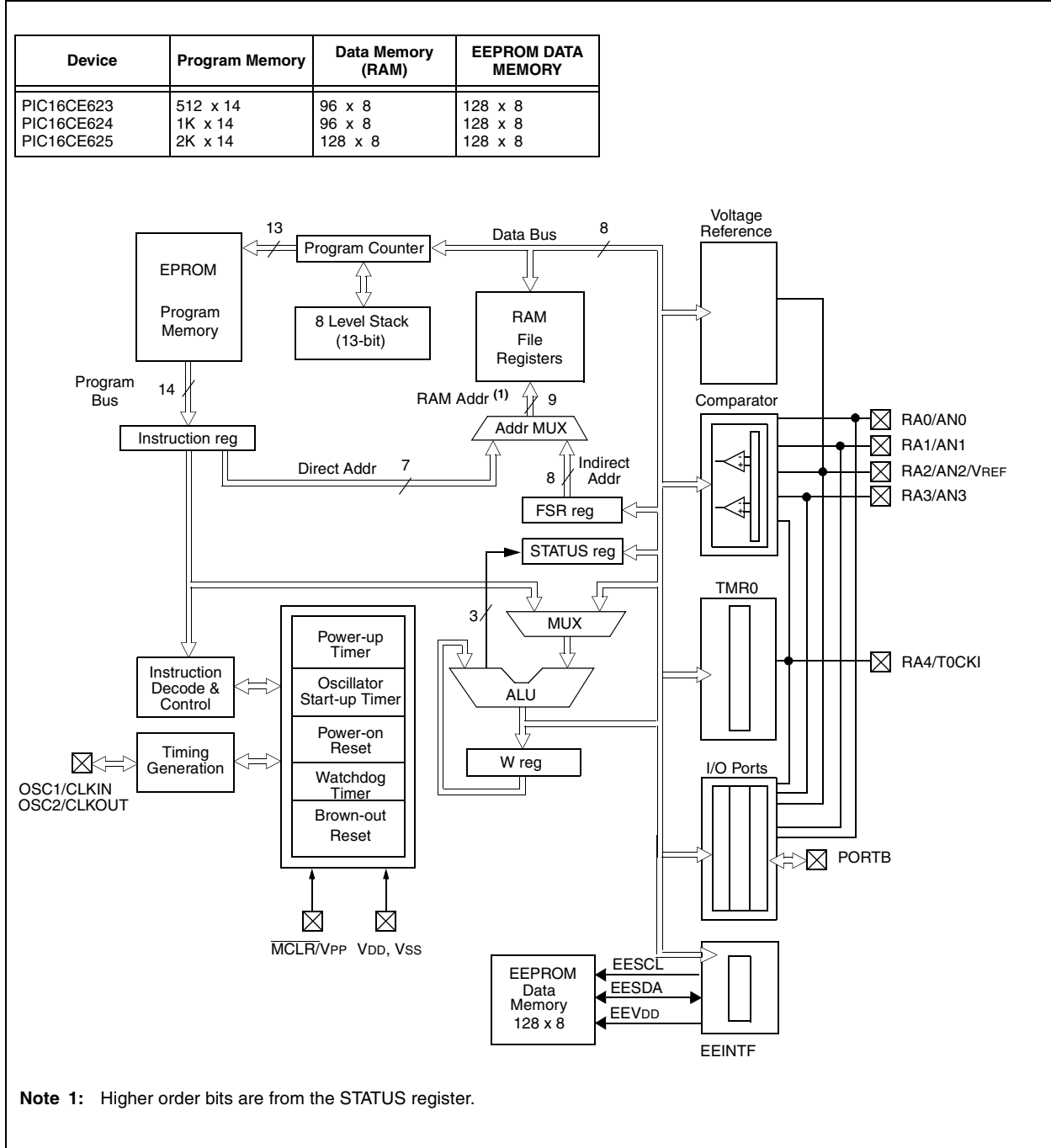


TABLE 3-1: PIC16CE62X PINOUT DESCRIPTION

Name	DIP/ SOIC Pin #	SSOP Pin #	I/O/P Type	Buffer Type	Description
OSC1/CLKIN	16	18	I	ST/CMOS	Oscillator crystal input/external clock source input.
OSC2/CLKOUT	15	17	O	—	Oscillator crystal output. Connects to crystal or resonator in crystal oscillator mode. In RC mode, OSC2 pin outputs CLKOUT which has 1/4 the frequency of OSC1, and denotes the instruction cycle rate.
MCLR/VPP	4	4	I/P	ST	Master clear (reset) input/programming voltage input. This pin is an active low reset to the device.
RA0/AN0 RA1/AN1 RA2/AN2/VREF RA3/AN3 RA4/TOCKI	17 18 1 2 3	19 20 1 2 3	I/O I/O I/O I/O I/O	ST ST ST ST ST	PORTA is a bi-directional I/O port. Analog comparator input Analog comparator input Analog comparator input or VREF output Analog comparator input /output Can be selected to be the clock input to the Timer0 timer/counter or a comparator output. Output is open drain type.
RB0/INT RB1 RB2 RB3 RB4 RB5 RB6 RB7	6 7 8 9 10 11 12 13	7 8 9 10 11 12 13	I/O I/O I/O I/O I/O I/O I/O	TTL/ST ⁽¹⁾ TTL TTL TTL TTL TTL/ST ⁽²⁾ TTL/ST ⁽²⁾	PORTB is a bi-directional I/O port. PORTB can be software programmed for internal weak pull-up on all inputs. RB0/INT can also be selected as an external interrupt pin. Interrupt on change pin. Interrupt on change pin. Interrupt on change pin. Serial programming clock. Interrupt on change pin. Serial programming data.
VSS	5	5,6	P	—	Ground reference for logic and I/O pins.
VDD	14	15,16	P	—	Positive supply for logic and I/O pins.

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

PIC16CE62X

3.1 Clocking Scheme/Instruction Cycle

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

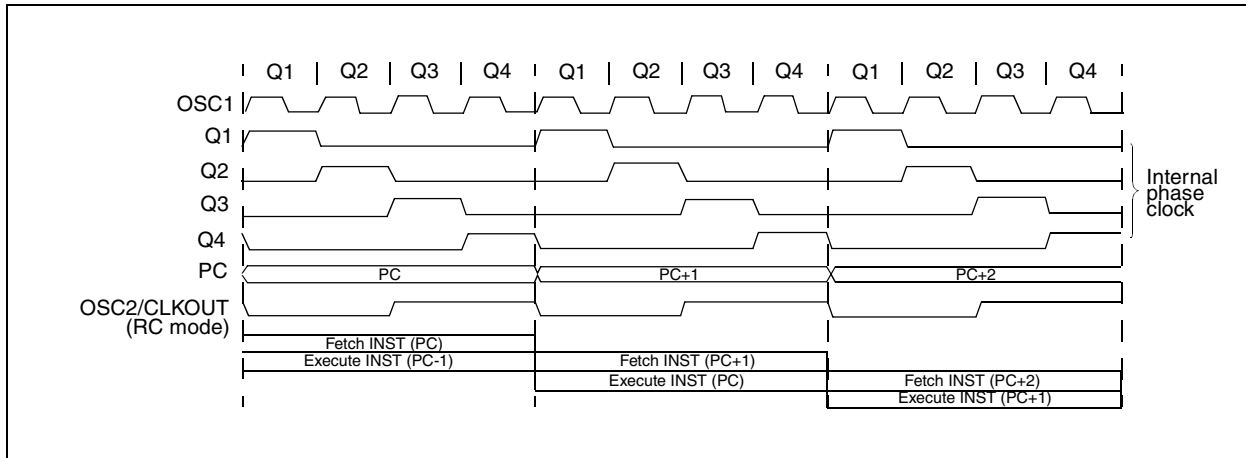
3.2 Instruction Flow/Pipelining

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

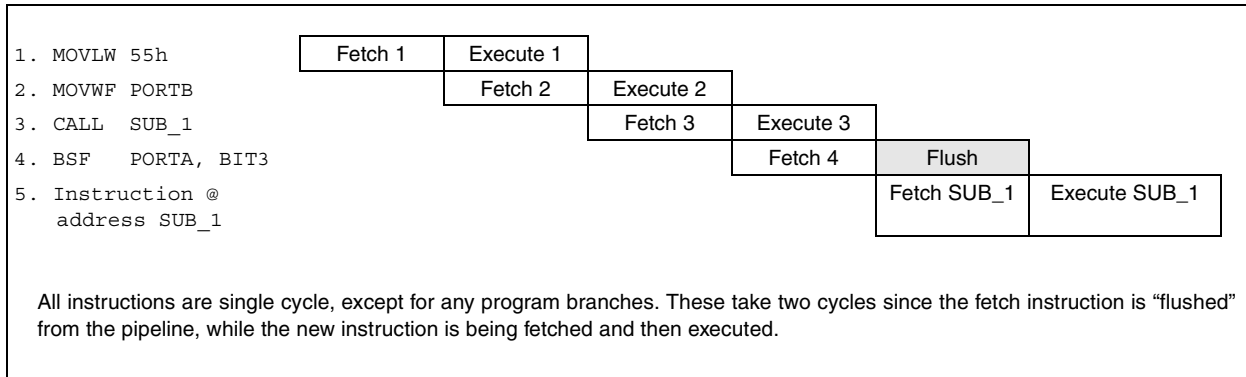
A fetch cycle begins with the program counter (PC) incrementing in Q1.

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

FIGURE 3-2: CLOCK/INSTRUCTION CYCLE



EXAMPLE 3-1: INSTRUCTION PIPELINE FLOW



4.0 MEMORY ORGANIZATION

4.1 Program Memory Organization

The PIC16CE62X has a 13-bit program counter capable of addressing an 8K x 14 program memory space. Only the first 512 x 14 (0000h - 01FFh) for the PIC16CE623, 1K x 14 (0000h - 03FFh) for the PIC16CE624 and 2K x 14 (0000h - 07FFh) for the PIC16CE625 are physically implemented. Accessing a location above these boundaries will cause a wrap-around within the first 512 x 14 space (PIC16CE623) or 1K x 14 space (PIC16CE624) or 2K x 14 space (PIC16CE625). The reset vector is at 0000h and the interrupt vector is at 0004h (Figure 4-1, Figure 4-2, Figure 4-3).

FIGURE 4-1: PROGRAM MEMORY MAP AND STACK FOR THE PIC16CE623

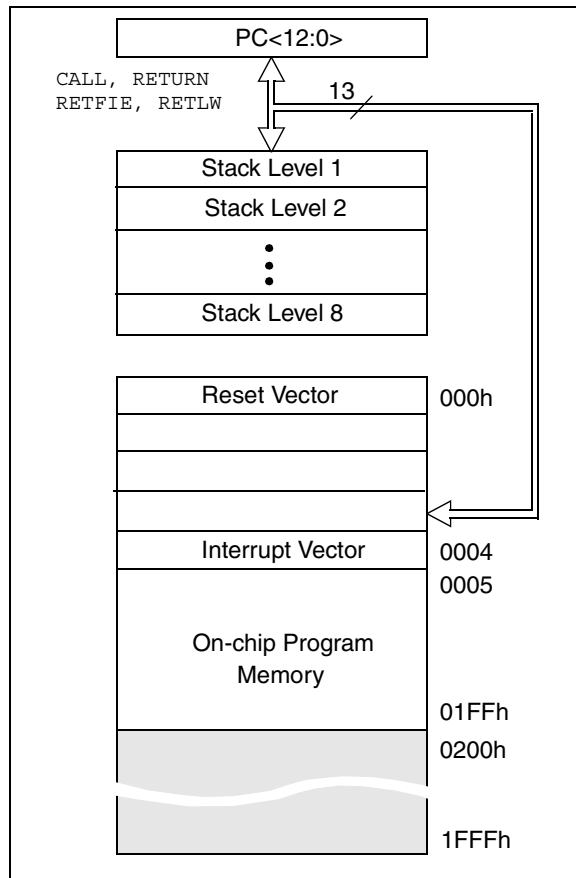


FIGURE 4-2: PROGRAM MEMORY MAP AND STACK FOR THE PIC16CE624

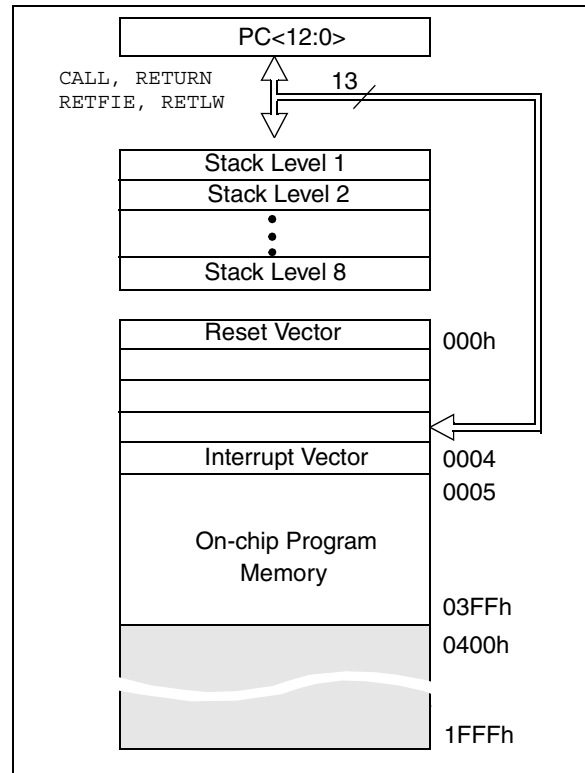
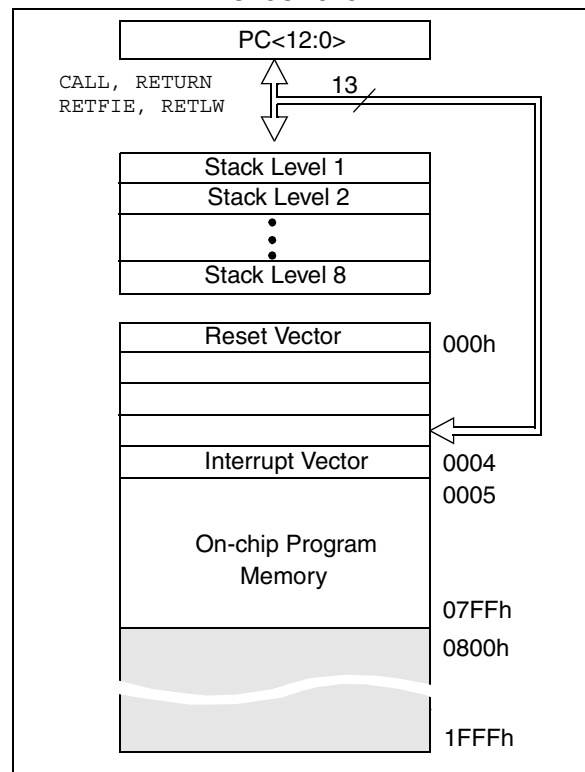


FIGURE 4-3: PROGRAM MEMORY MAP AND STACK FOR THE PIC16CE625



PIC16CE62X

4.2 Data Memory Organization

The data memory (Figure 4-4 and Figure 4-5) is partitioned into two Banks which contain the General Purpose Registers and the Special Function Registers. Bank 0 is selected when the RP0 bit is cleared. Bank 1 is selected when the RP0 bit (STATUS <5>) is set. The Special Function Registers are located in the first 32 locations of each Bank. Register locations 20-7Fh (Bank0) on the PIC16CE623/624 and 20-7Fh (Bank0) and A0-BFh (Bank1) on the PIC16CE625 are General Purpose Registers implemented as static RAM. Some special purpose registers are mapped in Bank 1. In all three microcontrollers, address space F0h-FFh (Bank1) is mapped to 70-7Fh (Bank0) as common RAM.

4.2.1 GENERAL PURPOSE REGISTER FILE

The register file is organized as 96 x 8 in the PIC16CE623/624 and 128 x 8 in the PIC16CE625. Each is accessed either directly or indirectly through the File Select Register FSR (Section 4.4).

PIC16CE62X

4.2.2 SPECIAL FUNCTION REGISTERS

The Special Function Registers are registers used by the CPU and peripheral functions for controlling the desired operation of the device (Table 4-1). 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” functions are described in this section. Those related to the operation of the peripheral features are described in the section of that peripheral feature.

TABLE 4-1: SPECIAL REGISTERS FOR THE PIC16CE62X

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on POR 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	Timer0 Module's Register								xxxx xxxx	uuuu uuuu
02h	PCL	Program Counter's (PC) Least Significant Byte								0000 0000	0000 0000
03h	STATUS	IRP ⁽²⁾	RP1 ⁽²⁾	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	—	—	—	RA4	RA3	RA2	RA1	RA0	---x 0000	---u 0000
06h	PORTB	RB7	RB6	RB5	RB4	RB3	RB2	RB1	RB0	xxxx xxxx	uuuu uuuu
07h	Unimplemented									—	—
08h	Unimplemented									—	—
09h	Unimplemented									—	—
0Ah	PCLATH	—	—	—	Write buffer for upper 5 bits of program counter			---	0000	---	0000
0Bh	INTCON	GIE	PEIE	TOIE	INTE	RBIE	TOIF	INTF	RBIF	0000 000x	0000 000u
0Ch	PIR1	—	CMIF	—	—	—	—	—	—	-0-- ----	-0-- ----
0Dh-1Eh	Unimplemented									—	—
1Fh	CMCON	C2OUT	C1OUT	—	—	CIS	CM2	CM1	CM0	00-- 0000	00-- 0000
Bank 1											
80h	INDF	Addressing this location uses contents of FSR to address data memory (not a physical register)								xxxx xxxx	xxxx xxxx
81h	OPTION	RBPU	INTEDG	TOCS	TOSE	PSA	PS2	PS1	PS0	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	—	—	—	TRISA4	TRISA3	TRISA2	TRISA1	TRISA0	---1 1111	---1 1111
86h	TRISB	TRISB7	TRISB6	TRISB5	TRISB4	TRISB3	TRISB2	TRISB1	TRISB0	1111 1111	1111 1111
87h	Unimplemented									—	—
88h	Unimplemented									—	—
89h	Unimplemented									—	—
8Ah	PCLATH	—	—	—	Write buffer for upper 5 bits of program counter			---	0000	---	0000
8Bh	INTCON	GIE	PEIE	TOIE	INTE	RBIE	TOIF	INTF	RBIF	0000 000x	0000 000u
8Ch	PIE1	—	CMIE	—	—	—	—	—	—	-0-- ----	-0-- ----
8Dh	Unimplemented									—	—
8Eh	PCON	—	—	—	—	—	—	POR	BOD	---- -0x	---- -uq
8Fh-9Eh	Unimplemented									—	—
90h	EEINTF	—	—	—	—	—	EESCL	EESDA	EEVDD	---- -111	---- -111
9Fh	VRCON	VREN	VROE	VRR	—	VR3	VR2	VR1	VR0	000- 0000	000- 0000

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 \overline{MCLR} reset, Brown-out Reset and Watchdog Timer Reset during normal operation.

Note 2: IRP & RPI bits are reserved; always maintain these bits clear.

4.2.2.1 STATUS REGISTER

The STATUS register, shown in Register 4-1, 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, 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 `000uu1uu` (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 bit. For other instructions, not affecting any status bits, see the "Instruction Set Summary".

Note 1: The IRP and RP1 bits (STATUS<7:6>) are not used by the PIC16CE62X and should be programmed as '0'. Use of these bits as general purpose R/W bits is NOT recommended, since this may affect upward compatibility with future products.

Note 2: 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.

REGISTER 4-1: STATUS REGISTER (ADDRESS 03H OR 83H)

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
bit7						bit0	

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

bit 7: **IRP:** The IRP bit is reserved on the PIC16CE62X, always maintain this bit clear.

bit 6:5 **RP<1:0>:** Register Bank Select bits (used for direct addressing)
11 = Bank 3 (180h - 1FFh)
10 = Bank 2 (100h - 17Fh)
01 = Bank 1 (80h - FFh)
00 = Bank 0 (00h - 7Fh)
Each bank is 128 bytes. The RP1 bit is reserved, always maintain this bit clear.

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/borrow bit (`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 (`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: 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 or low order bit of the source register.

PIC16CE62X

4.2.2.2 OPTION REGISTER

The OPTION register is a readable and writable register which contains various control bits to configure the TMR0/WDT prescaler, the external RB0/INT interrupt, TMR0 and the weak pull-ups on PORTB.

Note: To achieve a 1:1 prescaler assignment for TMR0, assign the prescaler to the WDT (PSA = 1).

REGISTER 4-2: OPTION 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
bit7	RBPU	INTEDG	T0CS	T0SE	PSA	PS2	PS1	PS0
								bit0

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

bit 7: **RBPU**: PORTB Pull-up Enable bit
 1 = PORTB pull-ups are disabled
 0 = PORTB pull-ups are enabled by individual port latch values

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: **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

4.2.2.3 INTCON REGISTER

The INTCON register is a readable and writable register which contains the various enable and flag bits for all interrupt sources except the comparator module. See Section 4.2.2.4 and Section 4.2.2.5 for a description of the comparator enable and flag bits.

Note: Interrupt flag bits get set when an interrupt condition occurs, regardless of the state of its corresponding enable bit or the global enable bit, GIE (INTCON<7>).

REGISTER 4-3: INTCON REGISTER (ADDRESS 0BH OR 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
GIE	PEIE	TOIE	INTE	RBIE	TOIF	INTF	RBIF
bit7							bit0
<div style="float: right; border: 1px solid black; padding: 5px; width: fit-content;"> <p>R = Readable bit W = Writable bit U = Unimplemented bit, read as '0' -n = Value at POR reset -x = Unknown at POR reset</p> </div> <p>bit 7: GIE: Global Interrupt Enable bit 1 = Enables all un-masked interrupts 0 = Disables all interrupts</p> <p>bit 6: PEIE: Peripheral Interrupt Enable bit 1 = Enables all un-masked peripheral interrupts 0 = Disables all peripheral interrupts</p> <p>bit 5: TOIE: TMR0 Overflow Interrupt Enable bit 1 = Enables the TMR0 interrupt 0 = Disables the TMR0 interrupt</p> <p>bit 4: INTE: RB0/INT External Interrupt Enable bit 1 = Enables the RB0/INT external interrupt 0 = Disables the RB0/INT external interrupt</p> <p>bit 3: RBIE: RB Port Change Interrupt Enable bit 1 = Enables the RB port change interrupt 0 = Disables the RB port change interrupt</p> <p>bit 2: TOIF: TMR0 Overflow Interrupt Flag bit 1 = TMR0 register has overflowed (must be cleared in software) 0 = TMR0 register did not overflow</p> <p>bit 1: INTF: RB0/INT External Interrupt Flag bit 1 = The RB0/INT external interrupt occurred (must be cleared in software) 0 = The RB0/INT external interrupt did not occur</p> <p>bit 0: RBIF: RB Port Change Interrupt Flag bit 1 = When at least one of the RB<7:4> pins changed state (must be cleared in software) 0 = None of the RB<7:4> pins have changed state</p>							

PIC16CE62X

4.2.2.4 PIE1 REGISTER

This register contains the individual enable bit for the comparator interrupt.

REGISTER 4-4: PIE1 REGISTER (ADDRESS 8CH)

U-0	R/W-0	U-0	U-0	U-0	U-0	U-0	U-0
—	CMIE	—	—	—	—	—	—

bit7 bit0

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

bit 6: **CMIE:** Comparator Interrupt Enable bit
1 = Enables the Comparator interrupt
0 = Disables the Comparator interrupt

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

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

4.2.2.5 PIR1 REGISTER

This register contains the individual flag bit for the comparator interrupt.

Note: Interrupt flag bits get 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 4-5: PIR1 REGISTER (ADDRESS 0CH)

U-0	R/W-0	U-0	U-0	U-0	U-0	U-0	U-0
—	CMIF	—	—	—	—	—	—

bit7 bit0

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

bit 6: **CMIF:** Comparator Interrupt Flag bit
1 = Comparator input has changed
0 = Comparator input has not changed

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

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

4.2.2.6 PCON REGISTER

The PCON register contains flag bits to differentiate between a Power-on Reset, an external $\overline{\text{MCLR}}$ reset, WDT reset or a Brown-out Reset.

Note: $\overline{\text{BOD}}$ is unknown on Power-on Reset. It must then be set by the user and checked on subsequent resets to see if $\overline{\text{BOD}}$ is cleared, indicating a brown-out has occurred. The $\overline{\text{BOD}}$ status bit is a "don't care" and is not necessarily predictable if the brown-out circuit is disabled (by programming BODEN bit in the configuration word).

REGISTER 4-6: PCON REGISTER (ADDRESS 8Eh)

U-0	U-0	U-0	U-0	U-0	U-0	R/W-0	R/W-0
—	—	—	—	—	—	$\overline{\text{POR}}$	$\overline{\text{BOD}}$
bit7						bit0	

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

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)

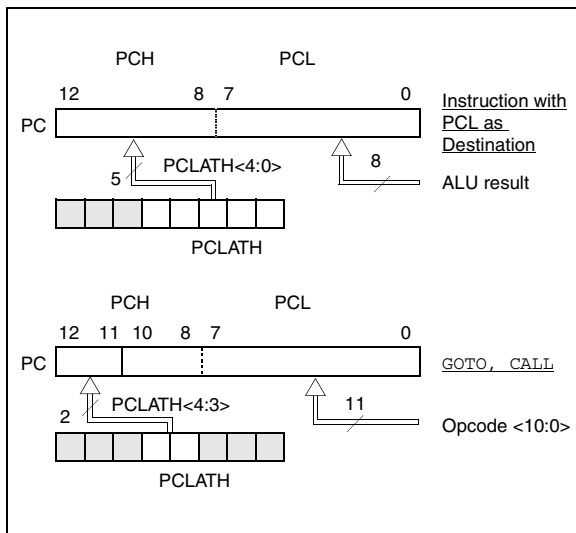
bit 0: **$\overline{\text{BOD}}$:** Brown-out Reset Status bit
 1 = No Brown-out Reset occurred
 0 = A Brown-out Reset occurred (must be set in software after a Brown-out Reset occurs)

PIC16CE62X

4.3 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 4-6 shows the two situations for the loading of the PC. The upper example in the figure shows how the PC is loaded on a write to PCL (PCLATH<4:0> → PCH). The lower example in the figure shows how the PC is loaded during a CALL or GOTO instruction (PCLATH<4:3> → PCH).

FIGURE 4-6: LOADING OF PC IN DIFFERENT SITUATIONS



4.3.1 COMPUTED GOTO

A computed GOTO is accomplished by adding an offset to the program counter (`ADDWF PCL`). When doing a table read using a computed GOTO method, care should be exercised if the table location crosses a PCL memory boundary (each 256 byte block). Refer to the application note, "Implementing a Table Read" (AN556).

4.3.2 STACK

The PIC16CE62X family has an 8 level deep x 13-bit wide hardware stack (Figure 4-2 and Figure 4-3). 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.

Note 2: There are no instruction/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.

4.4 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 register and the IRP bit (STATUS<7>), as shown in Figure 4-7. However, IRP is not used in the PIC16CE62X.

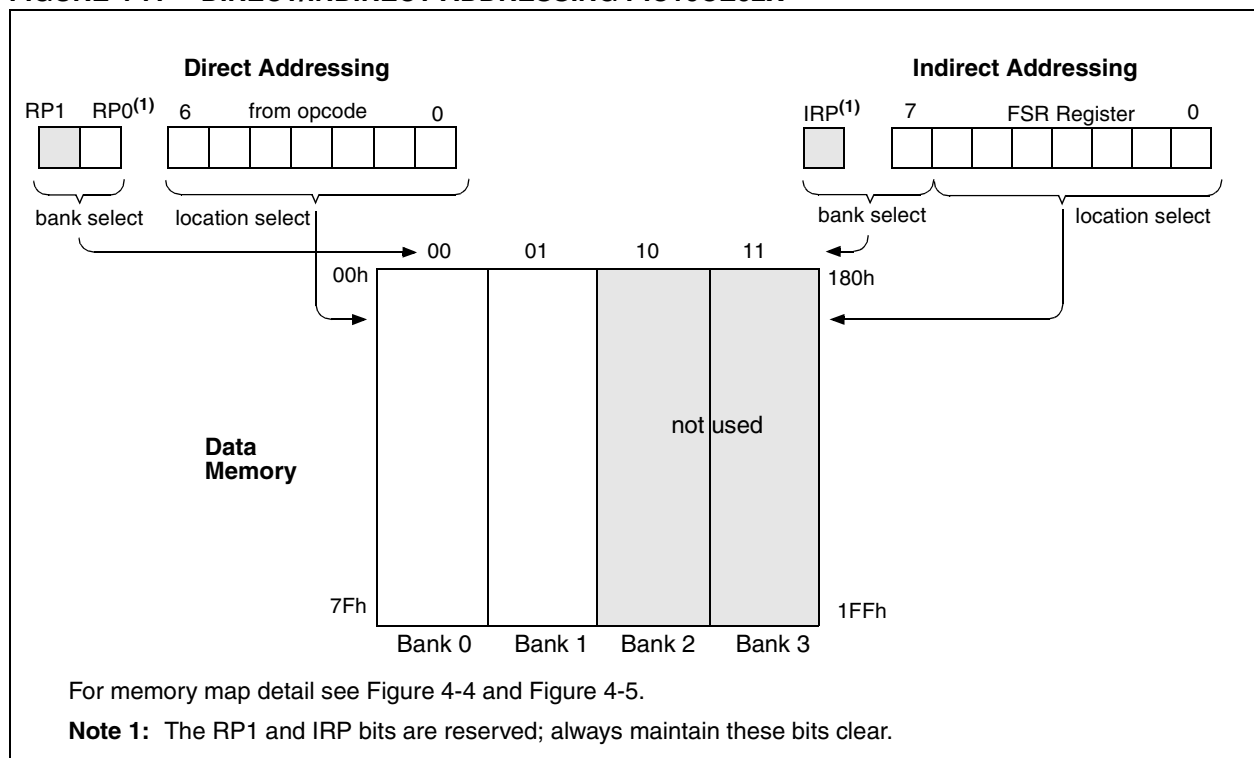
A simple program to clear RAM location 20h-2Fh using indirect addressing is shown in Example 4-1.

EXAMPLE 4-1: 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
                               ;yes continue
CONTINUE:
    
```

FIGURE 4-7: DIRECT/INDIRECT ADDRESSING PIC16CE62X



PIC16CE62X

NOTES:

5.0 I/O PORTS

The PIC16CE62X parts have two ports, PORTA and PORTB. Some pins for these I/O ports 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.

5.1 PORTA and TRISA Registers

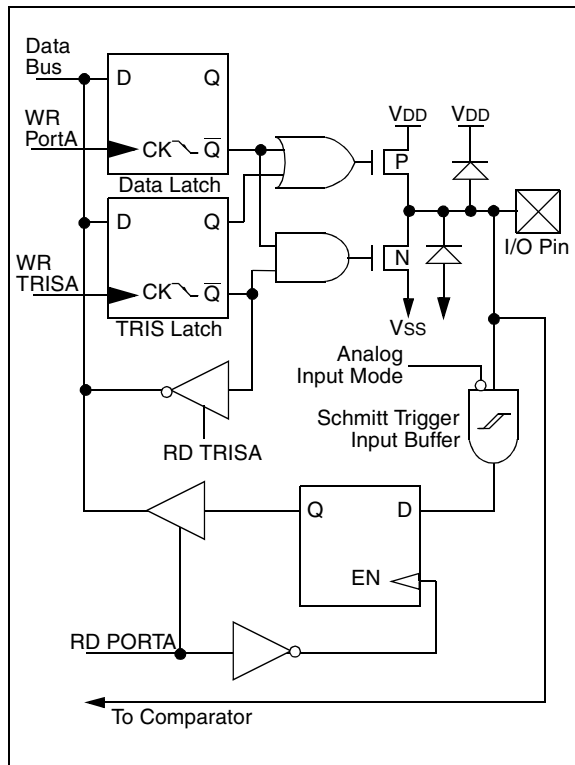
PORTA is a 5-bit wide latch. RA4 is a Schmitt Trigger input and an open drain output. Port RA4 is multiplexed with the T0CKI clock input. All other RA port pins have Schmitt Trigger input levels and full CMOS output drivers. All pins have data direction bits (TRIS registers), which can configure these pins as input or output.

A '1' in the TRISA register puts the corresponding output driver in a hi-impedance mode. A '0' in the TRISA register puts the contents of the output latch on the selected pin(s).

Reading the PORTA register reads the status of the pins, whereas writing to it will write to the port latch. All write operations are read-modify-write operations. So a write to a port implies that the port pins are first read, then this value is modified and written to the port data latch.

The PORTA pins are multiplexed with comparator and voltage reference functions. The operation of these pins are selected by control bits in the CMCON (Comparator Control Register) register and the VRCON (Voltage Reference Control Register) register. When selected as a comparator input, these pins will read as '0's.

FIGURE 5-1: BLOCK DIAGRAM OF RA<1:0> PINS



Note: On reset, the TRISA register is set to all inputs. The digital inputs are disabled and the comparator inputs are forced to ground to reduce excess current consumption.

TRISA controls the direction of the RA pins, even when they are being used as comparator inputs. The user must make sure to keep the pins configured as inputs when using them as comparator inputs.

The RA2 pin will also function as the output for the voltage reference. When in this mode, the VREF pin is a very high impedance output. The user must configure TRISA<2> bit as an input and use high impedance loads.

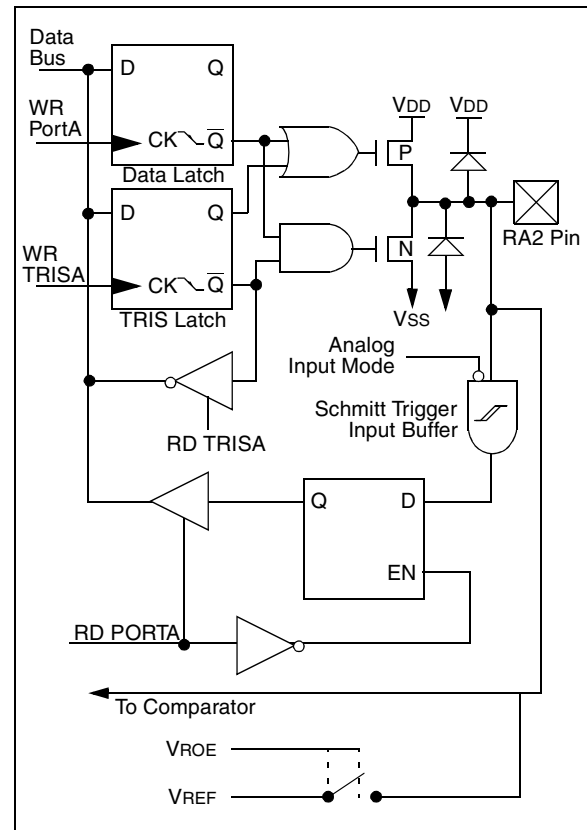
In one of the comparator modes defined by the CMCON register, pins RA3 and RA4 become outputs of the comparators. The TRISA<4:3> bits must be cleared to enable outputs to use this function.

EXAMPLE 5-1: INITIALIZING PORTA

```

CLRWF PORTA           ;Initialize PORTA by setting
                       ;output data latches
MOVWLW 0X07           ;Turn comparators off and
MOVWF CMCON           ;enable pins for I/O
                       ;functions
BSF STATUS, RP0      ;Select Bank1
MOVWLW 0x1F           ;Value used to initialize
                       ;data direction
MOVWF TRISA           ;Set RA<4:0> as inputs
                       ;TRISA<7:5> are always
                       ;read as '0'.
    
```

FIGURE 5-2: BLOCK DIAGRAM OF RA2 PIN



PIC16CE62X

FIGURE 5-3: BLOCK DIAGRAM OF RA3 PIN

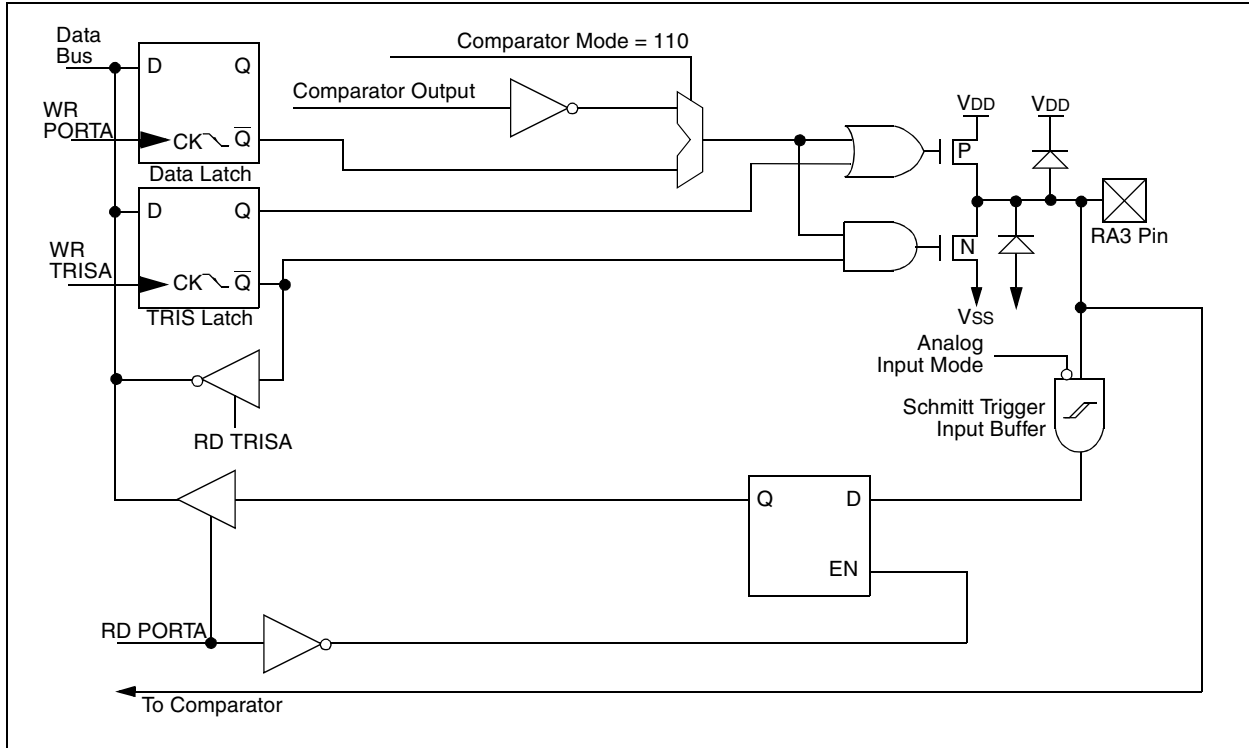


FIGURE 5-4: BLOCK DIAGRAM OF RA4 PIN

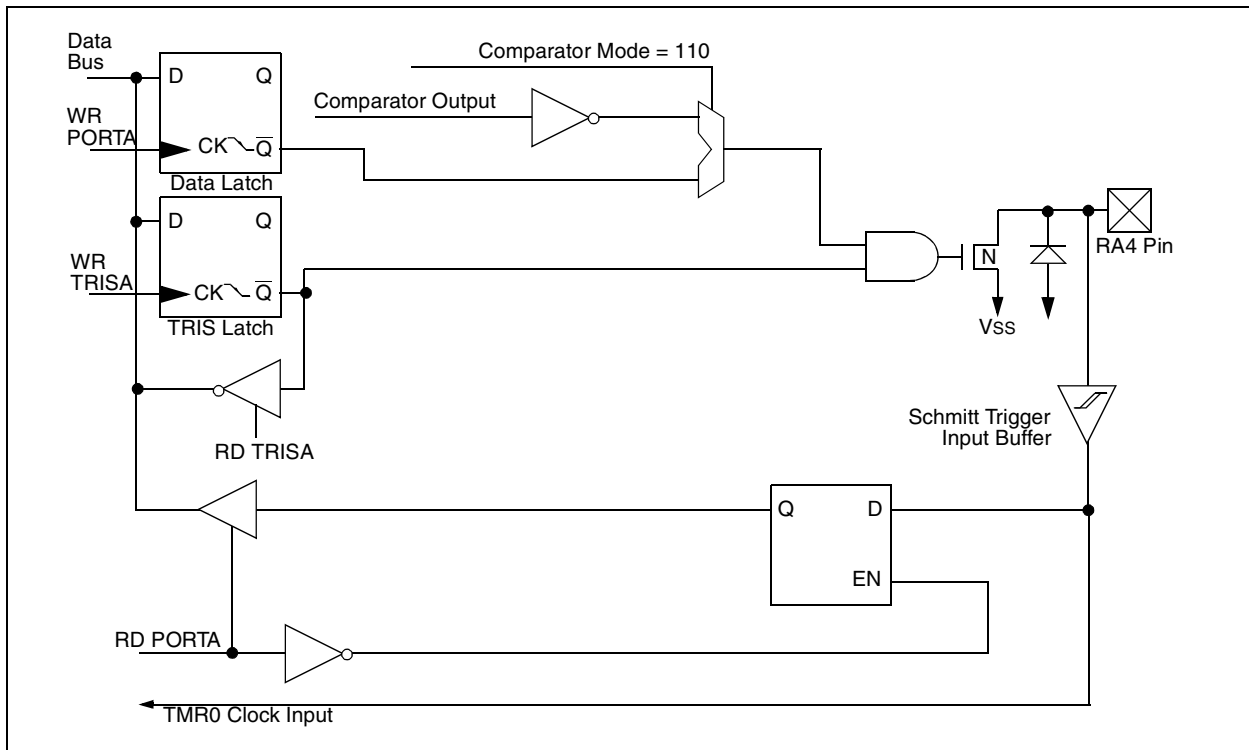


TABLE 5-1: PORTA FUNCTIONS

Name	Bit #	Buffer Type	Function
RA0/AN0	bit0	ST	Input/output or comparator input
RA1/AN1	bit1	ST	Input/output or comparator input
RA2/AN2/VREF	bit2	ST	Input/output or comparator input or VREF output
RA3/AN3	bit3	ST	Input/output or comparator input/output
RA4/T0CKI	bit4	ST	Input/output or external clock input for TMR0 or comparator output. Output is open drain type.

Legend: ST = Schmitt Trigger input

TABLE 5-2: 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	Value on All Other Resets
05h	PORTA	—	—	—	RA4	RA3	RA2	RA1	RA0	---x 0000	---u 0000
85h	TRISA	—	—	—	TRISA4	TRISA3	TRISA2	TRISA1	TRISA0	---1 1111	---1 1111
1Fh	CMCON	C2OUT	C1OUT	—	—	CIS	CM2	CM1	CM0	00-- 0000	00-- 0000
9Fh	VRCON	VREN	VROE	VRR	—	VR3	VR2	VR1	VR0	000- 0000	000- 0000

Legend: — = Unimplemented locations, read as '0', x = unknown, u = unchanged

Note: Shaded bits are not used by PORTA.