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

PIC16C64X & PIC16C66X

8-Bit EPROM Microcontrollers with Analog Comparators

Devices included in this data sheet:

- PIC16C641
- PIC16C642
- PIC16C661
- PIC16C662

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 x14	Data Memory x8
PIC16C641	2K	128
PIC16C642	4K	176
PIC16C661	2K	128
PIC16C662	4K	176

- Interrupt capability
- 8-level deep hardware stack
- Direct, Indirect and Relative addressing modes

Peripheral Features:

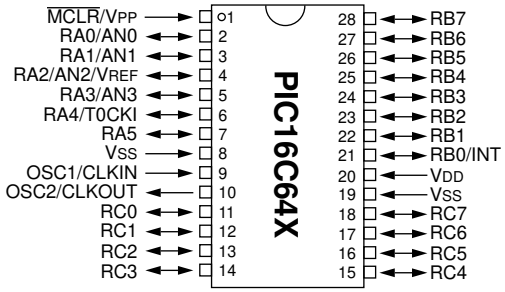
- Up to 33 I/O pins with individual direction control
- High current sink/source for direct LED drive
- Analog comparator module with:
  - Two analog comparators
  - Programmable on-chip voltage reference (V_{REF}) module
  - Programmable input multiplexing from device inputs and internal voltage reference
  - Comparator outputs can be output signals
- Timer0: 8-bit timer/counter with 8-bit programmable prescaler

Special Microcontroller Features:

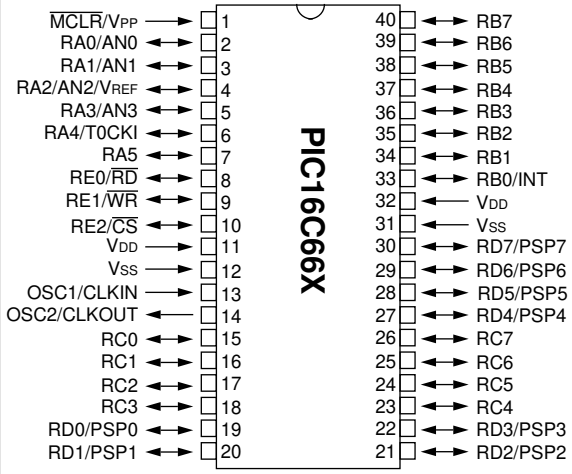
- Power-on Reset (POR)
- Power-up Timer (PWRT) and Oscillator Start-up Timer (OST)
- Brown-out Reset
- Watchdog Timer (WDT) with its own on-chip RC oscillator for reliable operation
- Programmable code protection
- Power saving SLEEP mode
- Selectable oscillator options
- Serial in-circuit programming (via two pins)

Pin Diagrams

PDIP, SOIC, Windowed CERDIP



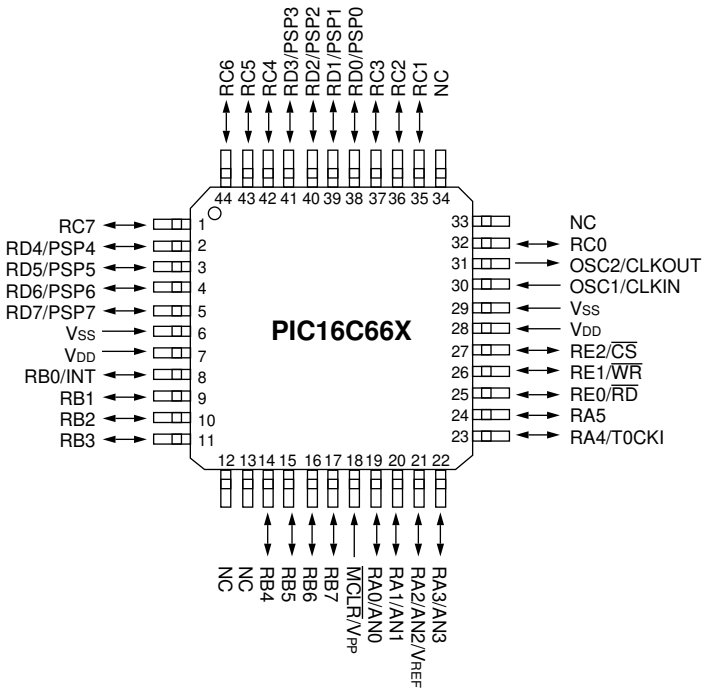
PDIP, Windowed CERDIP



- Four user programmable ID locations
- Program Memory Parity Error checking circuitry with Parity Error Reset (PER)
- **CMOS Technology:**
- Low-power, high-speed CMOS EPROM technology
- Fully static design
- Wide operating voltage range: 3.0V to 6.0V
- Commercial, Industrial and Automotive temperature ranges
- 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

Pin Diagrams (Cont.'d)

TQFP



PLCC

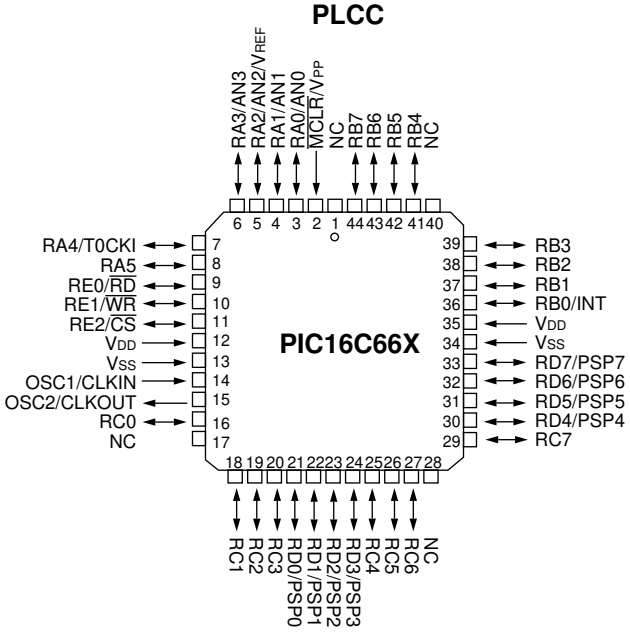


Table of Contents

To Our Valued Customers

We constantly strive to improve the quality of all our products and documentation. We have spent an exceptional amount of time to ensure that these documents are 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 use the reader response form in the back of this data sheet to inform us. We appreciate your assistance in making this a better document.

NOTES:

1.0 GENERAL DESCRIPTION

PIC16C64X & PIC16C66X devices are 28-pin and 40-pin EPROM-based members of the versatile PIC16CXXX family of low-cost, high-performance, CMOS, fully-static, 8-bit microcontrollers.

All PIC16/17 microcontrollers employ an advanced RISC architecture. The PIC16CXXX 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 the 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.

PIC16CXXX microcontrollers typically achieve a 2:1 code compression and a 4:1 speed improvement over other 8-bit microcontrollers in its class.

The PIC16C641 has 128 bytes of RAM and the PIC16C642 has 176 bytes of RAM. Both devices have 22 I/O pins, and an 8-bit timer/counter with an 8-bit programmable prescaler. In addition, they have two analog comparators with a programmable on-chip voltage reference module. Program Memory has internal parity error detection circuitry with a Parity Error Reset. The comparator module is ideally suited for applications requiring a low-cost analog interface (e.g., battery chargers, threshold detectors, white goods controllers, etc.).

The PIC16C661 has 128 bytes of RAM and the PIC16C662 has 176 bytes of RAM. Both devices have 33 I/O pins, and an 8-bit timer/counter with an 8-bit programmable prescaler. They also have an 8-bit Parallel Slave Port. In addition, the devices have two analog comparators with a programmable on-chip voltage reference module. Program Memory has internal parity error detection circuitry with a Parity Error Reset. The comparator module is ideally suited for applications requiring a low-cost analog interface (e.g., battery chargers, threshold detectors, white goods controllers, etc.).

PIC16CXXX devices have special features to reduce external components, thus reducing 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 saving. The user can wake-up the chip from SLEEP through several external and internal interrupts and resets.

A highly reliable Watchdog Timer (WDT) 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.

The PIC16CXXX series fit perfectly in applications ranging from 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 PIC16C64X & PIC16C66X very versatile.

1.1 Family and Upward Compatibility

Those users familiar with the PIC16C5X family of microcontrollers will realize that this is an enhanced version of the PIC16C5X architecture. Please refer to Appendix A for a detailed list of enhancements. Code written for PIC16C5X can be easily ported to the PIC16C64X & PIC16C66X (Appendix B).

1.2 Development Support

PIC16C64X & PIC16C66X devices are supported by the complete line of Microchip Development tools, including:

- MPLAB Integrated Development Environment including MPLAB-Simulator.
- MPASM Universal Assembler and MPLAB-C Universal C compiler.
- PRO MATE II and PICSTART Plus device programmers.
- PICMASTER In-circuit Emulator System
- *fuzzy*TECH-MP Fuzzy Logic Development Tools
- DriveWay Visual Programming Tool

Please refer to Section 11.0 for more details about these and other Microchip development tools.

TABLE 1-1: PIC16C64X & PIC16C66X DEVICE FEATURES

	Clock	Memory			Peripherals				Features			
	Maximum Frequency of Operation (MHz)	EPROM	Data Memory (bytes)	Program Memory	Timer Module(s)	Comparator(s)	Internal Reference Voltage	Parallel Slave Port	Interrupt Sources	I/O Pins	Voltage Range (Volts)	Brown-out Reset	Packages
PIC16C641	20	2K	128	TMR0	2	Yes	-	4	22	3.0-6.0	Yes	28-pin PDIP, SOIC, Windowed CDIP
PIC16C642	20	4K	176	TMR0	2	Yes	-	4	22	3.0-6.0	Yes	28-pin PDIP, SOIC, Windowed CDIP
PIC16C661	20	2K	128	TMR0	2	Yes	Yes	5	33	3.0-6.0	Yes	40-pin PDIP, Windowed CDIP; 44-pin PLCC, TQFP
PIC16C662	20	4K	176	TMR0	2	Yes	Yes	5	33	3.0-6.0	Yes	40-pin PDIP, Windowed CDIP; 44-pin PLCC, TQFP

All PIC16/17 Family devices have Power-on Reset, selectable Watchdog Timer, selectable code protect, and high I/O current capability.
All PIC16CXXX Family devices use serial programming with clock pin RB6 and data pin RB7.

2.0 PIC16C64X & PIC16C66X 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 Product Identification System page at the end of this data sheet. When placing orders, please use that page of the data sheet to specify the correct part number.

2.1 UV Erasable Devices

The UV erasable version, offered in 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® Plus and PRO MATE® II programmers both support programming of the PIC16C64X & PIC16C66X.

2.2 One-Time-Programmable (OTP) Devices

The availability of OTP devices is especially useful for customers who need 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-Turnaround-Production (QTP) Devices

Microchip offers a QTP Programming Service for factory production orders. This service is made available for users who choose not to program a medium to high quantity of units and whose code patterns have stabilized. The devices are identical to the OTP devices but with all EPROM locations and 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-Turnaround-Production (SQTP℠) Devices

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

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

NOTES:

3.0 ARCHITECTURAL OVERVIEW

The high performance of the PIC16C64X & PIC16C66X devices can be attributed to a number of architectural features commonly found in RISC microprocessors. To begin with, the PIC16C64X & PIC16C66X use 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 an 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, which require two cycles.

The PIC16C641 and PIC16C661 both address 2K x 14 on-chip program memory while the PIC16C642 and PIC16C662 address 4K x 14. All program memory is internal.

PIC16C64X & PIC16C66X devices can directly or indirectly address their register files or data memory. All special function registers including the program counter are mapped in the data memory. These devices have 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 PIC16C64X & PIC16C66X simple yet efficient. In addition, the learning curve is reduced significantly.

PIC16C64X & PIC16C66X 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 $\overline{\text{Borrow}}$ and $\overline{\text{Digit}}$ $\overline{\text{Borrow}}$ out bit, respectively, bit in subtraction. See the `SUBLW` and `SUBWF` instructions for examples.

FIGURE 3-1: PIC16C641/642 BLOCK DIAGRAM

PIC16C641 has 2K x 14 Program Memory and 128 x 8 RAM

PIC16C642 has 4K x 14 Program Memory and 176 x 8 RAM

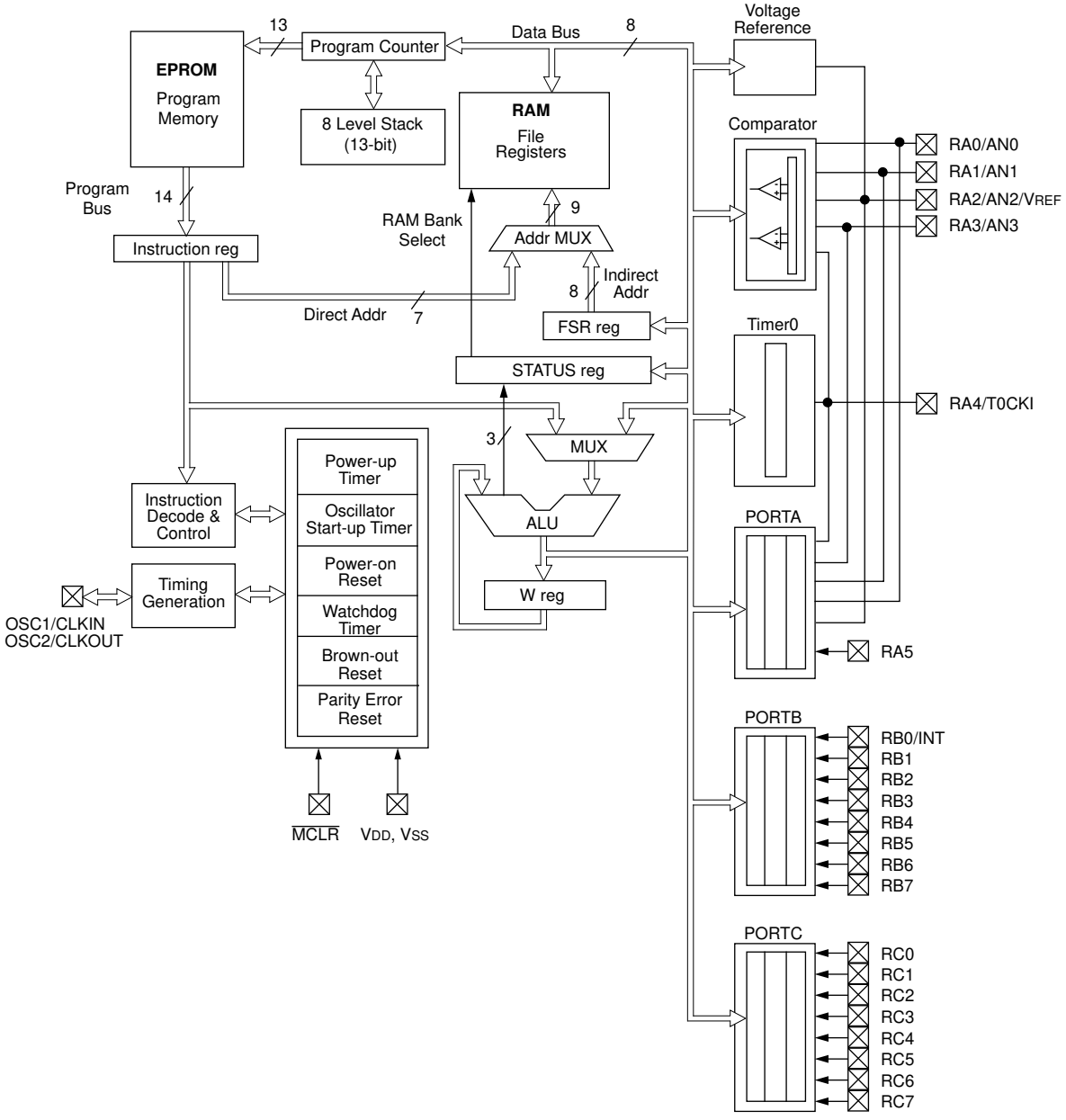


FIGURE 3-2: PIC16C661/662 BLOCK DIAGRAM

PIC16C661 has 2K x 14 Program Memory and 128 x 8 RAM

PIC16C662 has 4K x 14 Program Memory and 176 x 8 RAM

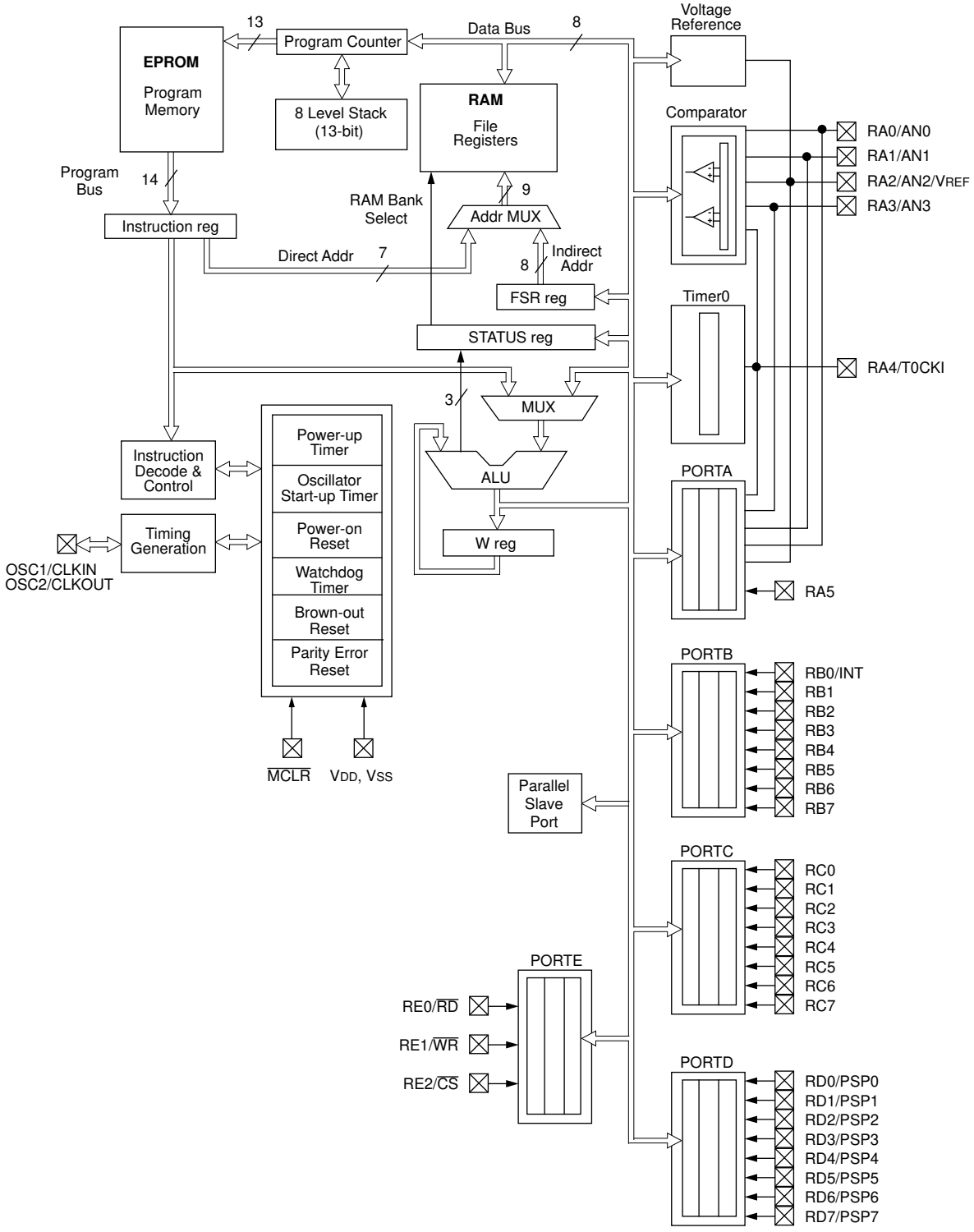


TABLE 3-1: PIC16C641/642 PINOUT DESCRIPTION

Name	Pin #	I/O/P Type	Buffer Type	Description
OSC1/CLKIN	9	I	ST/CMOS	Oscillator crystal input or external clock source input.
OSC2/CLKOUT	10	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.
$\overline{MCLR}$/VPP	1	I/P	ST	Master clear (reset) input or programming voltage input. This pin is an active low reset to the device.
				PORTA is a bi-directional I/O port.
RA0/AN0	2	I/O	ST	Analog comparator input.
RA1/AN1	3	I/O	ST	Analog comparator input.
RA2/AN2/VREF	4	I/O	ST	Analog comparator input or VREF output.
RA3/AN3	5	I/O	ST	Analog comparator input or comparator output.
RA4/T0CKI	6	I/O	ST	Can be selected to be the clock input to the Timer0 timer/counter or a comparator output. Output is open drain type.
RA5	7	I/O	ST	
				PORTB is a bi-directional I/O port. PORTB can be software programmed for internal weak pull-ups on all inputs.
RB0/INT	21	I/O	TTL/ST(1)	RB0 can also be selected as an external interrupt pin.
RB1	22	I/O	TTL	
RB2	23	I/O	TTL	
RB3	24	I/O	TTL	
RB4	25	I/O	TTL	Interrupt on change pin.
RB5	26	I/O	TTL	Interrupt on change pin.
RB6	27	I/O	TTL/ST(2)	Interrupt on change pin. Serial programming clock.
RB7	28	I/O	TTL/ST(2)	Interrupt on change pin. Serial programming data.
				PORTC is a bi-directional I/O port.
RC0	11	I/O	ST	
RC1	12	I/O	ST	
RC2	13	I/O	ST	
RC3	14	I/O	ST	
RC4	15	I/O	ST	
RC5	16	I/O	ST	
RC6	17	I/O	ST	
RC7	18	I/O	ST	
VSS	8,19	P	—	Ground reference for logic and I/O pins.
VDD	20	P	—	Positive supply for logic and I/O pins.

Legend: O = output, I = input, TTL = TTL input, I/O = input/output, — = not used, P = power, 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 3-2: PIC16C661/662 PINOUT DESCRIPTION

Name	DIP Pin #	QFP Pin #	PLCC Pin #	I/O/P Type	Buffer Type	Description
OSC1/CLKIN	13	30	14	I	ST/CMOS	Oscillator crystal input or external clock source input.
OSC2/CLKOUT	14	31	15	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.
$\overline{MCLR}$/VPP	1	18	2	I/P	ST	Master clear (reset) input or programming voltage input. This pin is an active low reset to the device.
						PORTA is a bi-directional I/O port.
RA0/AN0	2	19	3	I/O	ST	Analog comparator input.
RA1/AN1	3	20	4	I/O	ST	Analog comparator input.
RA2/AN2/VREF	4	21	5	I/O	ST	Analog comparator input or VREF output.
RA3/AN3	5	22	6	I/O	ST	Analog comparator input or comparator output.
RA4/T0CKI	6	23	7	I/O	ST	Can be selected to be the clock input to the Timer0 timer/counter or a comparator output. Output is open drain type.
RA5	7	24	8	I/O	ST	
						PORTB is a bi-directional I/O port. PORTB can be software programmed for internal weak pull-ups on all inputs.
RB0/INT	33	8	36	I/O	TTL/ST(1)	RB0 can also be selected as an external interrupt pin.
RB1	34	9	37	I/O	TTL	
RB2	35	10	38	I/O	TTL	
RB3	36	11	39	I/O	TTL	
RB4	37	14	41	I/O	TTL	Interrupt on change pin.
RB5	38	15	42	I/O	TTL	Interrupt on change pin.
RB6	39	16	43	I/O	TTL/ST(2)	Interrupt on change pin. Serial programming clock.
RB7	40	17	44	I/O	TTL/ST(2)	Interrupt on change pin. Serial programming data.
						PORTC is a bi-directional I/O port.
RC0	15	32	16	I/O	ST	
RC1	16	35	18	I/O	ST	
RC2	17	36	19	I/O	ST	
RC3	18	37	20	I/O	ST	
RC4	23	42	25	I/O	ST	
RC5	24	43	26	I/O	ST	
RC6	25	44	27	I/O	ST	
RC7	26	1	29	I/O	ST	

Legend: O = output, I = input, TTL = TTL input; I/O = input/output, — = not used; P = power, 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.
3: This buffer is a Schmitt Trigger input when configured as a general purpose I/O and a TTL input when used in the Parallel Slave Port Mode (for interfacing to a microprocessor port).

Name	DIP Pin #	QFP Pin #	PLCC Pin #	I/O/P Type	Buffer Type	Description
						PORTD can be a bi-directional I/O port or parallel slave port for interfacing to a microprocessor bus.
RD0/PSP0	19	38	21	I/O	ST/TTL(3)	
RD1/PSP1	20	39	22	I/O	ST/TTL(3)	
RD2/PSP2	21	40	23	I/O	ST/TTL(3)	
RD3/PSP3	22	41	24	I/O	ST/TTL(3)	
RD4/PSP4	27	2	30	I/O	ST/TTL(3)	
RD5/PSP5	28	3	31	I/O	ST/TTL(3)	
RD6/PSP6	29	4	32	I/O	ST/TTL(3)	
RD7/PSP7	30	5	33	I/O	ST/TTL(3)	
						PORTE is a bi-directional I/O port.
RE0/$\overline{RD}$	8	25	9	I/O	ST/TTL(3)	RE0/$\overline{RD}$ read control for parallel slave port.
RE1/$\overline{WR}$	9	26	10	I/O	ST/TTL(3)	RE1/$\overline{WR}$ write control for parallel slave port.
RE2/$\overline{CS}$	10	27	11	I/O	ST/TTL(3)	RE2/$\overline{CS}$ select control for parallel slave port.
VSS	12,31	6,29	13,34	P	—	Ground reference for logic and I/O pins.
VDD	11,32	7,28	12,35	P	—	Positive supply for logic and I/O pins.
NC	—	12,13, 33,34	1,17 28,40	—	—	Not Connected.

Legend: O = output, I = input, TTL = TTL input; I/O = input/output, — = not used; P = power, 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.
3: This buffer is a Schmitt Trigger input when configured as a general purpose I/O and a TTL input when used in the Parallel Slave Port Mode (for interfacing to a microprocessor port).

3.1 Clocking Scheme/Instruction Cycle

The clock input (from OSC1) 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 3-3.

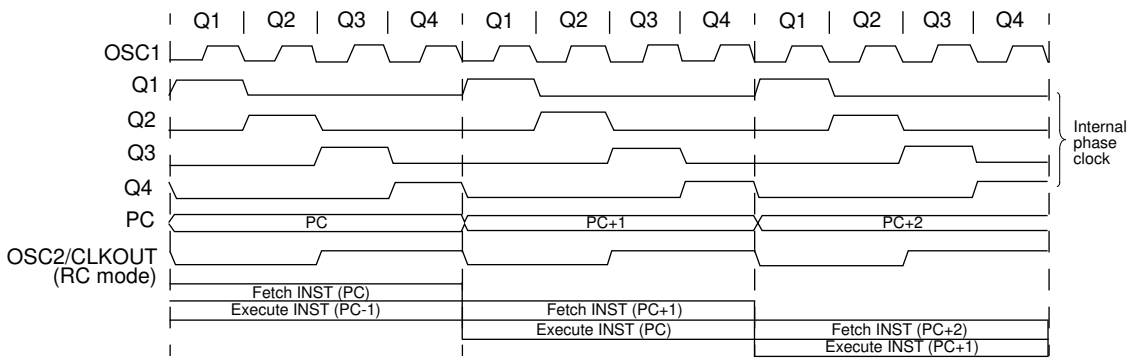
3.2 Instruction Flow/Pipelining

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 (e.g., `GOTO`) then two cycles are required to complete the instruction (Example 3-1).

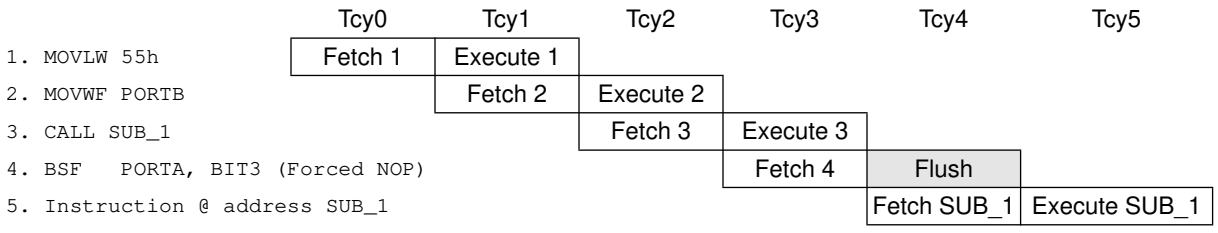
A fetch cycle begins with the program counter (PC) incrementing in Q1.

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

FIGURE 3-3: CLOCK/INSTRUCTION CYCLE



EXAMPLE 3-1: INSTRUCTION PIPELINE FLOW



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

NOTES:

4.0 MEMORY ORGANIZATION

4.1 Program Memory Organization

The PIC16C64X & PIC16C66X have a 13-bit program counter capable of addressing an 8K x 14 program memory space. For the PIC16C641 and PIC16C661 only the first 2K x 14 (0000h - 07FFh) is physically implemented. For the PIC16C642 and PIC16C662 only the first 4K x 14 (0000h - 0FFh) is physically implemented. Accessing a location above the 2K or 4K boundary will cause a wrap-around. The reset vector is at 0000h and the interrupt vector is at 0004h (Figure 4-1 and Figure 4-2). See Section 4.4 for Program Memory paging.

FIGURE 4-1: PIC16C641/661 PROGRAM MEMORY MAP AND STACK

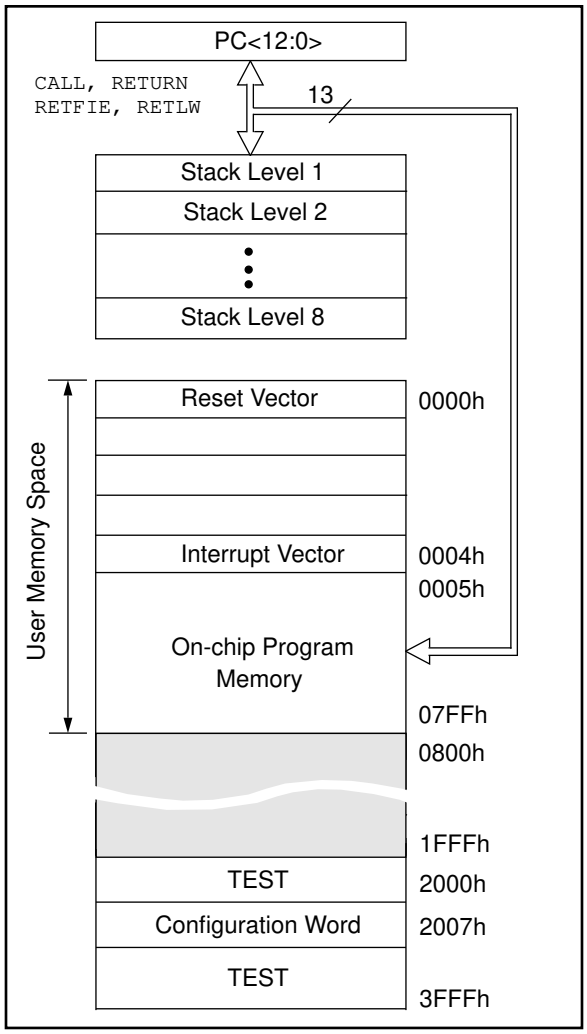
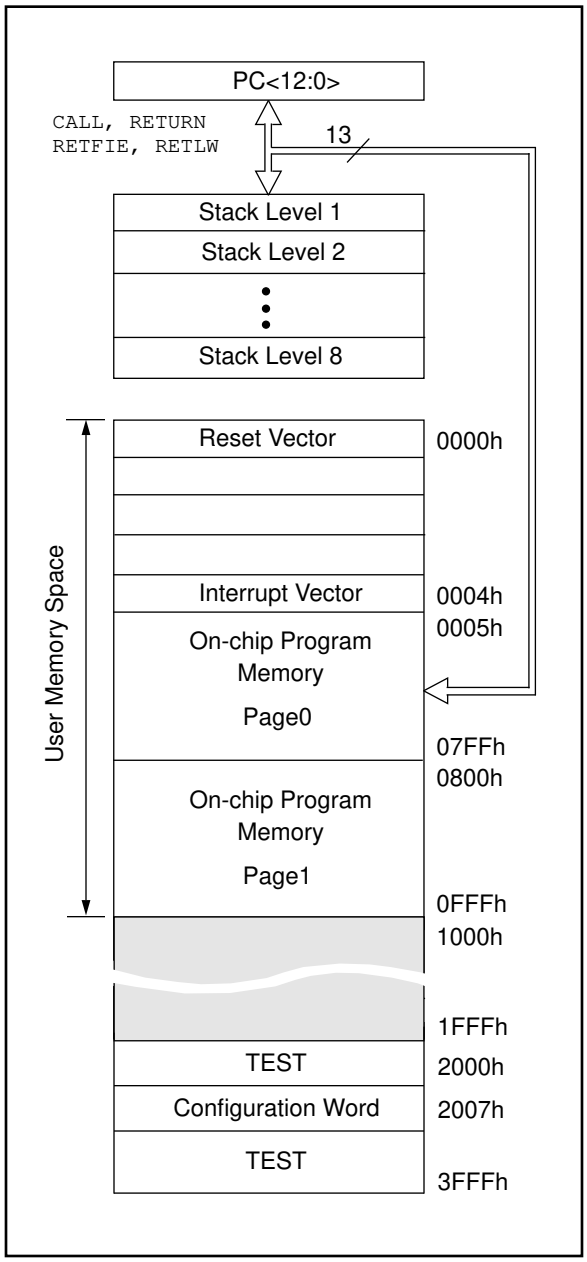


FIGURE 4-2: PIC16C642/662 PROGRAM MEMORY MAP AND STACK



4.2 Data Memory Organization

The data memory (Figure 4-4) is partitioned into two banks which contain the general purpose registers and the special function registers. Bank 0 is selected when bit RP0 (STATUS<5>) is cleared. Bank 1 is selected when the RP0 bit is set. The Special Function Registers are located in the first 32 locations of each Bank. Register locations A0h-EFh (Bank 1) are general purpose registers implemented as static RAM. Some special function registers are mapped in Bank 1.

4.2.1 GENERAL PURPOSE REGISTER FILE

The register file is organized as 176 x 8 for the PIC16C642/662, and 128 x8 for the PIC16C641/661. Each is accessed either directly, or indirectly through the File Select Register FSR (Section 4.5).

FIGURE 4-3: PIC16C641/661 DATA MEMORY MAP

File Address	Bank 0	Bank 1	File Address
00h	INDF(1)	INDF(1)	80h
01h	TMR0	OPTION	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	PORTD(2)	TRISD(2)	88h
09h	PORTE(2)	TRISE(2)	89h
0Ah	PCLATH	PCLATH	8Ah
0Bh	INTCON	INTCON	8Bh
0Ch	PIR1	PIE1	8Ch
0Dh			8Dh
0Eh		PCON	8Eh
0Fh			8Fh
10h			90h
11h			91h
12h			92h
13h			93h
14h			94h
15h			95h
16h			96h
17h			97h
18h			98h
19h			99h
1Ah			9Ah
1Bh			9Bh
1Ch			9Ch
1Dh			9Dh
1Eh			9Eh
1Fh	CMCON	VRCON	9Fh
20h	General Purpose Register	General Purpose Register	A0h
			BFh
			C0h
			EFh
		Mapped in Page 0	F0h
7Fh			FFh

Unimplemented data memory locations, read as '0'.

Note 1: Not a physical register.
2: Not implemented on the PIC16C641.

FIGURE 4-4: PIC16C642/662 DATA MEMORY MAP

File Address	Bank 0	Bank 1	File Address
00h	INDF(1)	INDF(1)	80h
01h	TMR0	OPTION	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	PORTD(2)	TRISD(2)	88h
09h	PORTE(2)	TRISE(2)	89h
0Ah	PCLATH	PCLATH	8Ah
0Bh	INTCON	INTCON	8Bh
0Ch	PIR1	PIE1	8Ch
0Dh			8Dh
0Eh		PCON	8Eh
0Fh			8Fh
10h			90h
11h			91h
12h			92h
13h			93h
14h			94h
15h			95h
16h			96h
17h			97h
18h			98h
19h			99h
1Ah			9Ah
1Bh			9Bh
1Ch			9Ch
1Dh			9Dh
1Eh			9Eh
1Fh	CMCON	VRCON	9Fh
20h	General Purpose Register	General Purpose Register	A0h
			EFh
		Mapped in Bank 0	F0h
7Fh			FFh

Unimplemented data memory locations, read as '0'.

Note 1: Not a physical register.
2: Not implemented on the PIC16C642.

4.2.2 SPECIAL FUNCTION REGISTERS

The special function registers are registers used by the CPU and Peripheral Modules for controlling the desired operation of the device (Table 4-1). These registers are static RAM.

The special function 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 FUNCTION REGISTERS

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on POR, BOR, PER	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	—	—	PORTA Data Latch when written: PORTA pins when read						--xx 0000	--xu 0000
06h	PORTB	PORTB Data Latch when written: PORTB pins when read								xxxx xxxx	uuuu uuuu
06h	PORTC	PORTC Data Latch when written: PORTC pins when read								xxxx xxxx	uuuu uuuu
06h	PORTD(3)	PORTD Data Latch when written: PORTD pins when read								xxxx xxxx	uuuu uuuu
06h	PORTE(3)	—	—	—	—	—	RE2	RE1	RE0	---- -xxx	---- -uuu
0Ah	PCLATH	—	—	—	Write buffer for upper 5 bits of program counter					---0 0000	---0 0000
0Bh	INTCON	GIE	PEIE	T0IE	INTE	RBIE	T0IF	INTF	RBIF	0000 000x	0000 000u
0Ch	PIR1	PSPIF(4)	CMIF	—	—	—	—	—	—	00-- ----	00-- ----
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	$\overline{RBPU}$	INTEDG	T0CS	T0SE	PSA	PS2	PS1	PS0	1111 1111	1111 1111
82h	PCL	Program Counter's (PC) Least Significant Byte								0000 0000	0000 0000
83h	STATUS	IRP(2)	RP1(2)	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	—	—	PORTA Data Direction Register						--11 1111	--11 1111
86h	TRISB	PORTB Data Direction Register								1111 1111	1111 1111
86h	TRISC	PORTC Data Direction Register								1111 1111	1111 1111
86h	TRISD(3)	PORTD Data Direction Register								1111 1111	1111 1111
86h	TRISE(3)	IBF	OBF	IBOV	PSPMODE	—	TRISE2	TRISE1	TRISE0	0000 -111	0000 -111
8Ah	PCLATH	—	—	—	Write buffer for upper 5 bits of program counter					---0 0000	---0 0000
8Bh	INTCON	GIE	PEIE	T0IE	INTE	RBIE	T0IF	INTF	RBIF	0000 000x	0000 000x
8Ch	PIE1	PSPIE(4)	CMIE	—	—	—	—	—	—	00-- ----	00-- ----
8Dh	Unimplemented									–	–
8Eh	PCON	MPEEN	—	—	—	—	$\overline{PER}$	$\overline{POR}$	$\overline{BOR}$	u--- -qqq	u--- -uuu
8Fh-9Eh	Unimplemented									–	–
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 and Watchdog Timer Reset during normal operation.
2: The IRP and RP1 bits are reserved, always maintain these bits clear.
3: The PORTD, PORTE, TRISD, and TRISE registers are not implemented on the PIC16C641/642.
4: Bits PSPIE and PSPIF are reserved on the PIC16C641/642, always maintain these bits clear.

4.2.2.1 STATUS REGISTER

The STATUS register, shown in Figure 4-5, 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 reserved on the PIC16C64X & PIC16C66X and should be maintained clear. 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 $\overline{Borrow}$ and $\overline{Digit\ Borrow}$ out bit, respectively, in subtraction. See the `SUBLW` and `SUBWF` instructions for examples.

FIGURE 4-5: 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	$\overline{TO}$	$\overline{PD}$	Z	DC	C
bit7							bit0

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

bit 7: **IRP:** Register Bank Select bit (used for indirect addressing)
1 = Bank 2, 3 (100h - 1FFh)
0 = Bank 0, 1 (00h - FFh)
Bit IRP is reserved on the PIC16C64X & PIC16C66X, always maintain this bit clear.

bit 6-5: **RP1:RP0**: 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. Bit RP1 is reserved on the PIC16C64X & PIC16C66X, 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/$\overline{borrow}$ 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 (`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 $\overline{borrow}$ 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 or low order bit of the source register.

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.

FIGURE 4-6: 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
$\overline{\text{RBPU}}$	INTEDG	T0CS	T0SE	PSA	PS2	PS1	PS0
bit7							bit0

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

bit 7: **$\overline{\text{RBPU}}$**: 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: **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

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 non-peripheral interrupt sources.

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>).

FIGURE 4-7: 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
GIE	PEIE	T0IE	INTE	RBIE	T0IF	INTF	RBIF
bit7							bit0

R= Readable bit
W= Writable bit
U= Unimplemented bit, read as '0'
- 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: **T0IE**: 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: **T0IF**: TMR0 Overflow Interrupt Flag bit
1 = TMR0 register overflowed (must be cleared in software)
0 = TMR0 register did not overflow

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

bit 0: **RBIF**: RB Port Change Interrupt Flag bit
1 = When at least one of the RB7:RB4 pins changed state (See Section 5.2 to clear interrupt)
0 = None of the RB7:RB4 pins have changed state

4.2.2.4 PIE1 REGISTER

This register contains the individual enable bits for the comparator and Parallel Slave Port interrupts.

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

R/W-0	R/W-0	U-0	U-0	U-0	U-0	U-0	U-0
PSPIE(1)	CMIE	—	—	—	—	—	—
bit7							bit0

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

bit 7: **PSPIE(1):** Parallel Slave Port Read/Write Interrupt Enable bit
1 = Enables the PSP read/write interrupt
0 = Disables the PSP read/write interrupt

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

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

Note 1: Bit PSPIE is reserved on the PIC16C641/642, always maintain this bit clear.

4.2.2.5 PIR1 REGISTER

This register contains the individual flag bits for the comparator and Parallel Slave Port interrupts.

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.

FIGURE 4-9: PIR1 REGISTER (ADDRESS 0Ch)

R/W-0	R/W-0	U-0	U-0	U-0	U-0	U-0	U-0
PSPIF(1)	CMIF	—	—	—	—	—	—
bit7							bit0

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

bit 7: **PSPIF(1):** Parallel Slave Port Interrupt Flag bit
1 = A read or write operation has taken place (must be cleared in software)
0 = No read or write operation has taken place

bit 6: **CMIF**: Comparator Interrupt Flag bit
1 = Comparator input has changed (must be cleared in software)
0 = Comparator input has not changed

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

Note 1: Bit PSPIF is reserved on the PIC16C641/642, always maintain this bit clear.