

I B M PC/XT

体 系 结 构

袁 保 玑 编写

中国科学院计算所公司

一九八八年四月

PDG

写在前面

七十年代中后期，美国 IBM 公司选用 Intel 公司生产的 CPU 芯汽，研制了它的十六位微型机——IBM PC（个人计算机）系列。

在 IBM 公司的个人计算机系列中，具有代表性的机型是：基本型 IBM PC、扩展型 IBM PC/XT 和增强型 IBM PC/AT，它们以先进的技术和优良的性能价格比赢得了广大用户的欢迎，特别是增强型 IBM PC/XT 及其兼容机，在我国具有非常广泛的市场。

本书拟向读者介绍 IBM PC/XT 的体系结构方案，其目的是期望广大的微机用户能够在使用微机的同时，进一步从使用的角度瞭解微机的硬件结构和工作原理，从而提高使用微机的水平，对于需要用汇编语言来编写程序的用户，体系结构更是不可缺少的知识。目前，IBM 公司又推出了新的微机系列 PS/2（个人系统/2'），它们以 Intel 80286 和 80386 作为 CPU，与 PC 系列相比，PS/2 速度更快、功能更强，安装与维护更为简便，性能价格比亦更佳。PS/2 系列与 PC 系列具有向上兼容性，因此，深入瞭解 IBM PC/XT 的体系结构亦将会为学习和使用新一代的微机系列打下良好的基础。

现代的计算机是一个由软件和硬件共同配合而组成的整体，这种配合的基础必然在体系结构中有所反映，因此，本书在介绍体系结构的同时，特别注意了软件与硬件的功能划分界面，以加深读者在这方面的印象。

目 录

写在前面

第一章 系统配置及物理结构.....	(1)
第一节 系统配置.....	(2)
第二节 系统物理结构概述.....	(4)
第二章 信息结构.....	(6)
第一节 字长.....	(7)
第二节 通用寄存器组.....	(7)
第三节 信息存取方式.....	(11)
第四节 编址方案和访问存储器地址的形成.....	(18)
第五节 数据表示.....	(26)
第六节 指令系统.....	(28)
第七节 代码系统.....	(44)
第三章 部件工作方式与控制方式.....	(45)
第一节 CPU 结构.....	(46)
第二节 中断系统.....	(46)
第三节 CPU 对外部设备的控制.....	(55)
第四章 指令功能介绍.....	(65)
结束语	

第一章 系统配置及物理结构

早在 1945 年第一台电子计算机研制期间,就由普林斯顿高级研究学院的冯·诺依曼等人在美国对数字计算机作了理论性的考察,并在此基础上对计算机的结构作了总结,他们所提出的结构设计思想被认为是计算机的“经典结构”,称作“冯·诺依曼结构”。但是对这种结构却一直未曾有十分严格的定义,大致说来,它有以下四个要点:

1. 计算机应由运算器、控制器、存贮器、输入设备和输出设备等五大部件组成。

2. 数据和程序均以二进制代码的形式不加区别地存放在存贮器中。

3. 由地址指定信息的存放位置。

4. 由一个指令计数器来控制指令的逐条执行。

数十年来,随着计算技术的不断发展,计算机的应用领域逐渐扩大,其结构形式亦趋于多种多样。但迄今为止,这种经典结构仍然具有强大的生命力,IBM——PC/XT 就是遵循此种经典结构的一个机型。按照习惯的叫法,我们把其中的运算器和控制器的组合称为“CPU (中央处理器)”,把能进行随机读写的存贮设备称为“内存贮器”,而把输入设备,输出设备和大容量的外存贮器称为“外部设备”。它们与电源、通风等设备一起,构成了计算机的硬件,在 PC/XT 机中,如图 1—1 所示,各大部件之间是通过总线来进行信息传递的。

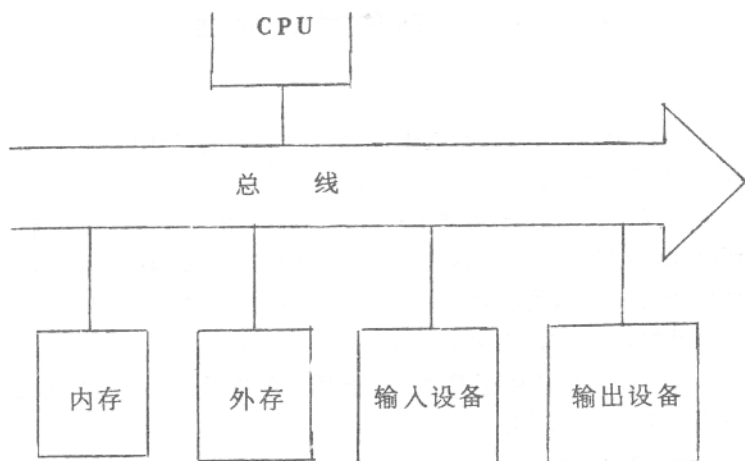


图 1 - 1 各大部件之间通过总线传递信息

本章将扼要地介绍 PC / XT 机的系统配置及其物理结构的示意图。

第一节 系统配置

PC / XT 机的标准系统配置包括系统部件、键盘部件、IBM 单色显示器、行式打印机和系统扩充选件, 前面四部份是 PC / XT 的基本配置。

1. 系统部件

系统部件包括以下八个部分。

- 1) 16 位微处理器 8088 , 以及支持它工作的各种辅助

电路。

2) 16 位协处理器 8087。(可选)。

3) 44 KB 容量的ROM

ROM 为“只读存储器”，在PC/XT 运行时，可以对其内容进行“读出”操作，但不能写入。

ROM 中存放的信息包括：

支持硬盘的基本输入输出系统BIOS，其中支持硬盘的驱动程序的容量为4KB，其它驱动程序的容量为3KB。

上电自检程序，共2KB。

基本BASIC 解释程序，共32KB。

存放硬盘驱动程序的ROM 器件安装在硬盘适配卡上，其它程序的ROM 器件安装在主机系统板上。

4) 128 KB 或 256 KB 容量的RAM

RAM 为“随机读写存储器”，它作为内存存储器，可存放操作系统和用户程序的各种信息。

5) 5.25 吋软盘驱动器一台

可对5.25 吋的双面盘进行读写，在DOS 2.0 以上版本操作系统的控制下，格式化后的存储容量可达360 KB，其信息分布如下：2面/片，40道/面，9扇区/道，512字节/扇区。

6) 10 MB 容量的硬盘驱动器一台

硬盘驱动器含有两片5.25 吋的温盘，格式化后的存储容量可达10 MB 左右，其信息分布如下：2片/台，2面/片，306道/面，17扇区/道，512字节/扇区。

7) 八个系统扩充插槽

供插入软盘驱动器、硬盘驱动器、显示器、打印机等适配卡以及各种扩充选件之用。

8) 电源

2. 83 键的键盘部件

3. IBM 单色显示器

4. 行式打印机

5. 系统扩充选件, 计有:

1) RAM 可扩充到 640 KB

2) IBM 单色显示器和打印机适配卡

3) 彩色/图形适配卡

4) 5.25 吋软盘驱动器适配卡

5) 异步通讯适配卡

6) 打印机适配卡

7) 游戏控制适配卡

8) 网络适配卡

等等。

第二节 系统物理结构概述

从物理结构的角度来看, PC / XT 机所有的运算处理、存贮和输入输出接口电路等都集中在主机箱内的一块大底板(称为“系统板”)和各种被选用的适配卡上, 适配卡插入系统板上的系统扩充插槽, 使之与系统板构成一个整体而工作。显示器、键盘和打印机等各种外部设备通过电缆与主机箱中的接口电路相连接。

系统板是系统部件的核心, 它水平地安装在主机箱内, 其功能划分如图 1 - 2 所示:

主 振 荡 器 时 钟 同 步 控 制	8088微处理器 8087微处理器	直接存储器 访问控制 DMA	定时器 与 计数器	中断控 制电路
电 源 插 座	只读存储器 ROM		随机读写 存储器 RAM	
扬声控制器	输入/输出通道 (包括系统扩充插槽)			
键盘接口电路				

图 1 - 2 系统板功能划分示意图

8088微处理器包括运算器和控制器，是PC/XT中的CPU（中央处理部件）。

8087微处理器以协从处理器的身份出现，它用于处理高速的浮点运算，以弥补8088运算功能的不足。

支持8088工作的辅助电路有：主振荡器及时钟信号发生器8284A、DMA控制器8237A-5、定时器/计数器8253-5以及中断控制电路8259A等，它们的主要功能是：

主振荡器使8088和8087获得4.77 MHz（210 ns振荡周期）的时钟信号。

“DMA”是“直接存贮器访问”的意思，它为外部设备和内存之间提供了直接进行数据传送的通路，使得数据传输不再经过CPU中的累加器，从而提供了这些设备与CPU并行工作的可能性。8237A-5为四通道的DMA控制器，其中三个通道供输入输出总线使用（用于软盘、硬盘和网络适配卡），而另一个通道用于动态存贮器的刷新。

三路定时器、计数器分别用于存贮器刷新、实现日时钟和支持扬声器发声。

中断控制电路对八个外部硬件请求中断信号进行排队，

並选出当前最优一级请中信号送往CPU请求中断。

除上述器件外，系统板上还包括ROM、RAM（不包括扩充配置）、键盘和扬声器的控制电路及八个系统扩充插槽。

扩充插槽是具有62个插脚的插座，八个插槽的对应插脚相互连接与附加的控制电路一起形成输入输出总线，它是8088内部总线的扩充。各种输入输出设备和其它选件的适配卡均可插入插槽与输入输出总线发生联系。在输入输出总线上，最多可允许1024个端口地址，其中前512个地址供系统板上各种控制电路使用，后512个地址供适配卡使用。

系统板和各种适配卡可经由这些地址所指定的端口，通过输入输出总线与CPU及其它设备交换信息。

总线上可以传输的信息宽度是8位，因此，一个16位长的数据需要二次传输。在以下各章中，我们将要对各种功能进行更详细的说明。

第二章 信息结构

计算机的运算对象是诸如浮点数、定点数、十进制数和字符串等各种形式的数据，而运算过程的控制则是由一串有序的指令组成的各种程序，程序和数据均以二进制代码的形式存放在计算机内的各种存贮设备（如存贮器、寄存器、磁盘等）中。我们把它们统称为“信息”。把信息的各种指标，如信息的长度、类型、存取方式、表示方式、存放信息的各种存贮设备的编址方案以及输入输出代码系统等统称为“信息结构”。

信息是计算机的“粮食”，在机器中，源源不断的信息流被输入、存贮、传送、加工和输出，从而组成了整个计算

机的工作过程。如果没有信息的存在，计算机也就失去了运行的意义。

信息结构是确定机器功能最基本的因素。譬如说，数据的长度、类型、表示方式和指令系统所定义的各种数据运算方式基本上决定了运算器的重要功能；指令的长度、形式、地址宽度、信息存取方式和指令系统所定义的各种控制型指令的功能基本上决定了控制器的主要功能。而信息的长度、地址宽度、存贮设备编址方案和输入输出代码系统又在很大程度上决定了传输总线功能。因此，在了解一台机器的体系结构时，信息结构是一个重点。

IBM - PC / XT 机的信息结构是由 CPU 芯片 8088 所决定的。

第一节 字 长

我们通常所说的计算机字长，是指运算器中可以用一条指令处理的数据的（二进制）长度，它是计算机信息的一种基本长度。

在有关指令系统一节中我们将会看到 8088 指令所处理的数据可以有 8 位长和 16 位长两种，前者称为“字节处理”，后者称为“字处理”。因为它的运算器是 16 位长，所以它有一次性处理 16 位数据的能力，针对这种情况，我们称 PC / XT 机的字长为 16 位。

第二节 通用寄存器组

对应于 8 位和 16 位两种数据，8088 分别提供了两个寄存器组，每组有 8 个寄存器。

一、用于字节处理的 8 位寄存器组

这样的寄存器共有 8 个，成为一组。每个寄存器的长度为 8 位，它们的名字和地址编号为：

名字		地址编号	
AH	AL	100	000
BH	BL	111	011
CH	CL	101	001
DH	DL	110	010

这 8 个寄存器均可以作为累加器使用。所谓累加器，是指 CPU 内部的这样一种寄存器，它们在执行一条指令时，既存放操作数，又存放操作后的结果。AL 是一个比较特殊的累加器，被称为“8 位的主累加器”，在特定的指令形式中，可以隐含地（即不指名地）使用它。

二、用于字处理的 16 位通用寄存器组

所谓“通用寄存器”，一般是指既可存放数据，又可存放地址或地址增量的寄存器。在 8088 中，这样的寄存器亦有 8 个，成为一组，每个寄存器的长度为 16 位，它们的名字和地址编号为：

名字	编号	名字	编号
AX	000	BP	101
BX	011	SP	100
CX	001	SI	110
DX	010	DI	111

这八个通用寄存器的名字含义如下：

AX：累加器（ACCUMULATOR）

BX：基址寄存器（BASE）

CX：计数值寄存器（COUNT）

DX：数据寄存器（DATA）

SP: 栈指针 (STACK POINTER) 寄存器

BP: 基址指针 (BASE POINTER) 寄存器

SI: 源操作数变址器 (SOURCE INDEX)

DI: 目的操作数变址器 (DESTINATION INDEX)

这八个通用寄存器有以下四种用途:

1. 作为累加器

同 AL 一样, AX 被称为“16位的主累加器”。

2. 作为地址寄存器

指令中所指出的存储器地址或端口地址均为16位长, 这些16位长的地址可以存放在16位通用寄存器中, 每个地址指向一个8位宽度的存储位置。

在这些寄存器中, BX、BP 称为“基址寄存器”和“基址指针寄存器”, 它们经常用来存放内存信息块的基地址。图 2.1 是一个基地址的例子, 在RAM中的一个数据块, 它的块起始存放地址可以看作为一个基地址。当基地址发生改变时, 数据块就在RAM中浮动, 而数据块中每一元素的地址可以通过基地址和该元素在块中的相对地址相加后得到。

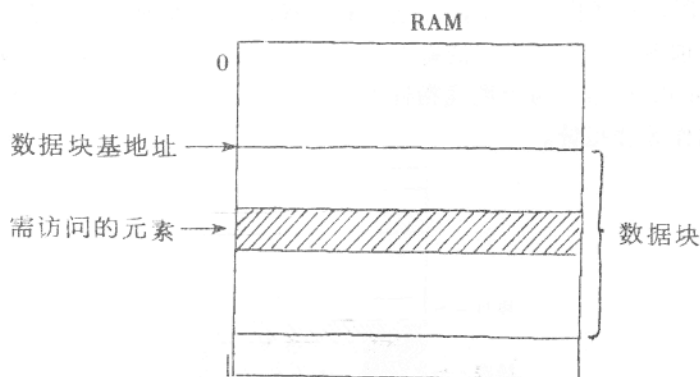


图 2 - 1 基地址的定义举例

SI、DI称为“变址寄存器”，它们经常用来存放可变地址中的地址改变量。例如，某数组 ARRAY 中有 100 个元素，如果在一个循环体中设有一条读取数组元素的指令，要求每循环一次取其中的一个元素，那末可以将指令中的元素地址设计为可变地址，地址改变量放在变址寄存器中，每循环一次，把变址寄存器的内容修改一次，以形成下一个元素的地址。

指令中的外部设备端口地址，规定放在 DX 寄存器中。

3. 作为计数器

在某些指令中，指定 CX 作为计数器使用。

4. 作为堆栈指针

指定一批地址连续的存贮区域作为存放信息的“仓库”，称为“堆栈”，其一端的地址是固定的，称为“栈底”，另一端的地址是浮动的，称为“栈顶”，栈内的信息都是固定长度的，存取信息均对栈顶而言，而且后进入堆栈的信息一定先被取出，故而称为“后进先出堆栈”。图 2 - 2 是后进先出堆栈的示意图。可以看到，在这样的堆栈中，当信息入栈时，栈顶向上移动，而当信息出栈时，栈顶则向下移动。栈顶地址存放在专用的寄存器中，供进栈、出栈时使用。这样的寄存器称为“堆栈指针”。在 PC/XT 中，用 SP 寄存器作为栈指针。

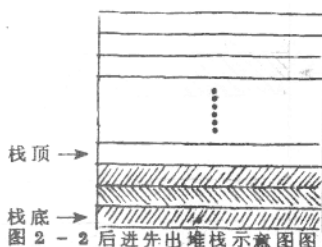


图 2 - 2 后进先出堆栈示意图

第三节 信息存取方式

在 PC/XT 机中, 采用“指定地址”和“后进先出堆栈”两种存取信息的方式。

一、指定地址

将各个可以存放信息的存贮位置(如寄存器、内存贮器等), 按一定的规则进行编号, 称为“地址号”, 可以用地址指定要进行存取的存贮位置, 然后再进行信息的存入和取出。

二、后进先出堆栈

可由程序员在 RAM 中自行建立堆栈, 数量不限, 可把每个堆栈看作为“一段信息”, 称为一个“栈段”。但是, 当前允许使用的堆栈只有一个, 称为“当前栈段”。一个堆栈的最大容量为 64KB。

PC/XT 机的堆栈, 栈底是地址较大的一端, 栈顶在地址较小的一端。由栈段寄存器 SS 和栈指针 SP 来定义堆栈和指出栈顶地址。它的标准形式如图 2-3 所示。栈段起始地址放在 SS 寄存器中, 栈顶和栈段起始地址之间的地址差则存放在 SP 寄存器中。每个栈段的容量不超过 64 KB。

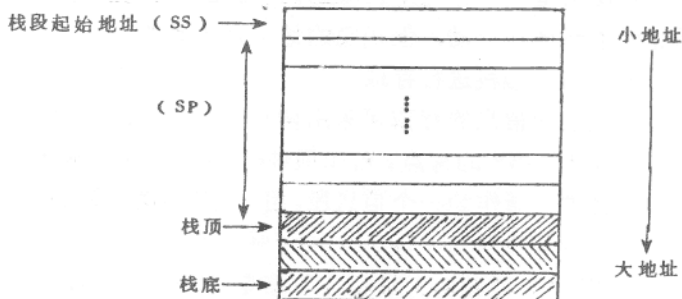


图 2-3 PC/XT 机的堆栈示意图

下面，我们将从两个方面来进一步描述堆栈。

1. 如何定义一个堆栈

送好 SS 和 SP 两个寄存器的内容，就定义了一个当前栈段。

堆栈一定是定义在 RAM 中的，而 RAM 的最大容量可达 640KB，因此，RAM 的地址要用 20 位来表示，但是，在 PC/XT 中，为了与运算器的长度相一致，所有的地址寄存器均为 16 位长。为了能在 SS 中存放一个 RAM 地址，规定栈段起始地址的最低 4 位一定为零，可以隐含地表示，而将较高的 16 位地址存放在 SS 中，栈段的容量不会超过 64 KB，因此栈顶和栈段起始地址之间的地址差用 16 位即可表示，它被存放在 SP 中，用 SS 的内容加上 SP 的内容，就可以得到栈顶所在的地址。

当栈段为“空”时，栈顶和栈底合一，因此，在初始定义堆栈时，SP 的内容实际上决定了栈底的地址。例如，我们将 0105H（H 代表十六进制数）送往 SS，将 0012H 送往 SP，这就定义了一个栈段，其起始地址为 01050H，其栈底地址为 $01050H + 0012H = 01062H$ 。

在使用堆栈的过程中，SP 的内容不断地被更改，栈顶也因此而不断地在浮动，但栈底的地址是不动的。

2. 如何对堆栈进行存取

对堆栈内信息的存取可采用两种方法来进行，一种是利用它的“后进先出”的特点，对栈顶进行“压入”和“弹出”，另一种是将堆栈看作为一个信息段，用一般的传送指令进行存取。

1) 对栈顶进行“压入”和“弹出”

CPU 中设计有两条堆栈存取指令 PUSH 和 POP，它们分别用于“将寄存器或内存中的信息压入堆栈”和“把栈顶

内容弹出到寄存器或内存”，每次交换信息的长度固定为16位。

图 2-4 是堆栈存取过程的示意图。

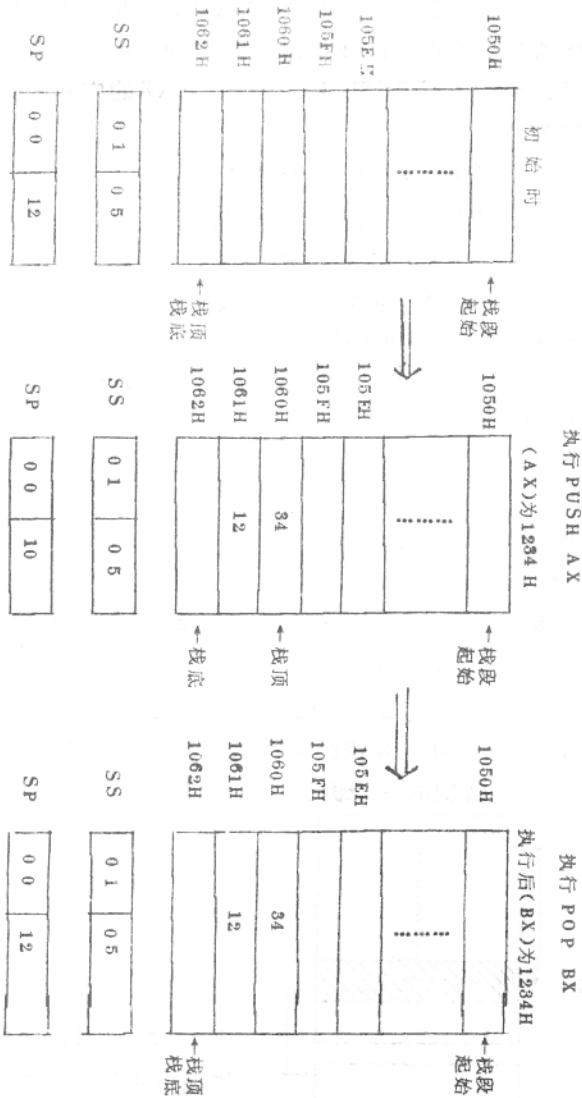


图 2-4 堆栈“压入”和“弹出”举例

图中第一步,先建立一个堆栈,其栈段起始地址为01050H,栈空时,栈顶与栈底合一,栈顶地址为 01062 H。

图中第二步,执行 PUSH AX 指令,将AX的内容压入堆栈,其动作为:

$(SP) - 1 \longrightarrow SP$
 $(AX) \text{ 高8位} \longrightarrow \text{栈顶}$
 $(SP) - 1 \longrightarrow SP$
 $(AX) \text{ 低8位} \longrightarrow \text{栈顶}$

此时栈顶地址为 01060 H。

图中第三步,执行 POP BX 指令,将栈顶内容弹出到BX寄存器,其动作为:

栈顶内容 $\longrightarrow (BX) \text{ 低8位}$
 $(SP) + 1 \longrightarrow SP$
 栈顶内容 $\longrightarrow (BX) \text{ 高8位}$
 $(SP) + 1 \longrightarrow SP$

这样的操作结果,使AX的内容通过堆栈送往BX中。

2) 用特定的 MOV指令

若将存取目标在栈段中的相对位置放在BP寄存器内,那么利用MOV指令可以对栈段中任意位置进行信息存取。图2-5是这种情况的一个例子。

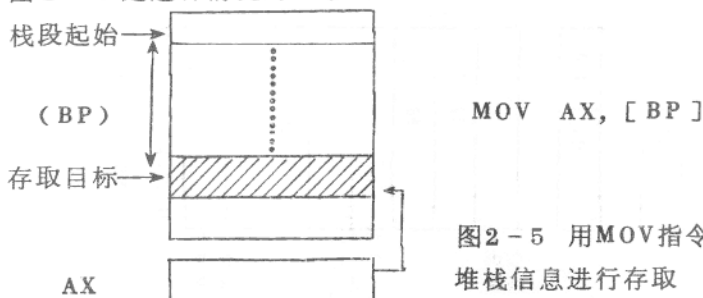


图2-5 用MOV指令对堆栈信息进行存取