微型计算机系列培训教程（第二册）

张载鸿　主编

微型机(PC 系列)系统功能教程

张昆藏　编

清 华 大 学 出 版 社

内 容 提 要

本书以汇编语言为工具，通过 40 个实例详细介绍了使用 IBM-PC 系列微机（PC、PC／XT、PC／AT）的 BIOS 中断功能和 DOS 系统功能的编程方法和技巧。并介绍了几种常用高级语言与汇编语言的混合编程技术。

全书共九章：中央处理器及指令系统，宏汇编程序的伪指令，汇编语言程序上机操作，系统层次结构，使用 DOS 和 BIOS 的字符设备 I／O 功能，使用 DOS 和 BIOS 的定时与发声功能，使用 DOS 的文件管理系统功能，使用 BIOS 的磁盘、串行 I／O 功能和扩充 BIOS 以及高级语言调用汇编子程序。还有习题解答、指令集、BIOS 软件中断和 DOS 系统功能四个附录。

本书选材精炼、论述清晰、实例丰富，是 PC 系列机应用培训班的良好教材，亦是高等院校计算机应用专业师生和软件开发人员的益友。

（京）新登字 158 号

微型机（PC 系列）系统功能教程

张昆藏 编

☆

清华大学出版社出版

北京 清华园

朝阳经纬印刷厂印刷

新华书店总店科技发行所发行

☆

开本：787×1092 1／16 印张：20.5 字数：521 千字

1992 年 8 月第 1 版 1992 年 8 月第 1 次印刷

印数：00001-10000

ISBN 7-302-01096-X／TP・411

定价：14.00 元

序

《微型计算机系列教程》和广大读者见面了。

本套系列教程的出版顺应了当今微型计算机发展的总趋势，可满足各行各业微型计算机应用人员学习、掌握PC系统的迫切需求，并对进一步开发应用微型机技术向深层次开拓起到强大的推动作用。

微型计算机在70年代末期由8位步入16位机的发展阶段，经过80年代整整十年的普及和应用，使微型计算机这一高科技领域中的"宠儿"触及了社会的各个角落，它为计算机发展历史作出了不可磨灭的功绩。在众多型号的微型计算机家族中，尤以IBM PC系列微型机，因其先进的结构设计和丰富的系统软件和实用软件能适应各种层次应用人员的要求，而成为过去十年间微型计算机工业事实上的生产标准和市场导向。

自80年代后期开始，微型计算机进入32位机的发展阶段，各类386机、486机相继占领个人计算机的市场。根据我国经济实力和发展趋势推测，在二十世纪的最后一个十年间，国内的微型计算机应用技术仍将与IBM PC系列兼容的各类PC机作为主要开发对象。

基于此，一套以强调基础性、实用性和系统性为编写宗旨的《微型计算机系列教程》在90年代的今天问世，乃是时代发展的要求。

本套系列教程以IBM PC系列兼容机为展开讨论的模式，本着由浅入深，从外到里的原则，按PC机层次结构一步步地引导读者了解、使用和掌握微型机系统的结构原理、操作和维护方法以及软件编程技术和硬件维修手段，最终达到PC系列开发应用的目的。

本套系列教程的主编张载鸿副教授制定了编写大纲，确立其编写风格，并对全套教程各册的内容作了全面的审阅和修改。参与各册教程编写的作者都是工作在微型计算机教学、科研、培训和维修第一线的教师和工程师，他们积累了丰富的实践经验。在各册编写中，始终贯彻"理论与实践相结合、系统与应用相结合、软件与硬件相结合"的方针，使本套教程具有"内容简炼、编排新颖、叙述清晰、实例丰富"的特色。各册教程每章后面附有习题，有的检验读者对内容理解的程度，有的可增长读者更多的知识。

愿所有的读者从中获得启迪和乐趣!

《微型计算机系列教程》分两批出版。首批出版的是前三册，其内容分列于下：

第一册书名是《微型机（PC系列）应用基础教程》，由王路敬执笔。

该册共分九章。从计算机基础知识入手，叙述微型机的结构特点、安装检测和日常维护；着重讲解中西文操作系统（PC-DOS和CC-DOS）的命令使用和上机操作，强调软盘和硬盘的使用技术；对汉化DBASEⅢ和常用的汉字输入软件、编辑软件、制表软件和WPS排版软件作了实用性的阐述，并具体指出数据库编程方法和技巧。

第二册书名是《微型机（PC系列）系统功能教程》由张昆藏执笔。

该册共分九章。从计算机核心CPU结构入手，在叙述指令系统的基础上，具体讲解汇编语言编程方法；PC系统提供的功能有数百个，包括DOS功能和BIOS功能两大类，通过数十个实例详细描述字符设备I/O、磁盘I/O、时钟设备I/O以及文件管理等功能，最后，讨论各种高级语言调用汇编子程序的编程技术。

第三册书名是《微型机（PC 系列）接口控制教程》，由张载鸿执笔。

该册共分九章。从计算机接口控制方式入手，在叙述微型机常用的程序查询、中断控制和 DMA 传输三种控制原理的基础上，讲解 PC 系列各个接口控制卡的硬件逻辑和软件编程的相互关系；具体指出对 I／O 接口芯片、键盘控制器、日时钟和实时钟、打印机控制卡、串行通信控制卡、彩色图形控制卡、软盘控制卡和硬盘控制卡的编程技术。

计划第二批出版的后二册，其内容安排如下：

第四册书名是《微型计算机硬件维修教程》。

该册从微型机常用的逻辑电路入手，在叙述微型机硬件结构的基础上，讲解系统板、显示器、打印机、内存储器、软盘子系统和硬盘子系统等部件的维修手段和方法，并通过实例分析常见故障的处置技术。

第五册书名是《微型计算机高级技术教程》。

该册从操作系统 DOS 结构剖析入手，公开 DOS 内核的全部功能（包括 DOS 保留功能），通过具体实例讲解 DOS 不可重入性及其对策，内存驻留程序 TSR 编程技术，进程管理和多任务接口，以及网络重定向技术等。

本套系列教程可用作各类微型机技术培训班的教材，也可作为大专院校师生和各行业人员学习微型计算机的参考用书。

本套系列教程是由中国科学院北京科海培训中心组织编写的，华根娣主任、夏非彼编辑对本套丛书的编写工作提出不少宝贵意见，在此表示衷心感谢。

由于本人水平有限，主编一套系列教程是初次尝试，难免有不当之处，谨请读者批评指正。

主编　张载鸿

于北京计算机学院

1992 年 2 月

前　言

本书是中国科学院北京科海培训中心组织编写的《微型计算机系列培训教程》的第二册。在第一册的基础上，本书将介绍宏汇编语言，以此为工具，通过40个实例详细介绍使用BIOS中断功能和DOS系统功能的编程方法和技术，最后介绍几种常用高级语言与汇编语言的混合编程技术。本书的另一目的是使读者对PC系列微机的系统层次结构有更深入的了解，以便为系列教材的以后各册的学习打下基础。

本书共分九章并有四个附录。可分为如下四个部分:

第一～第三章是基础部分。它介绍Intel 8088和80286微处理器及其寻址方式和指令系统，并以30个短例来说明各类指令的使用；接着介绍宏汇编程序的伪指令和伪操作符；最后介绍汇编语言程序的编辑、汇编、连接、装入运行等上机实际操作的过程以及程序的调试手段，并阐明.EXE文件与.COM文件的区别及程序段前缀结构等编程的必备知识。这三章相当于IBM PC汇编语言程序设计的简明教程，提供了5个程序范例，每章附有习题可供读者练习。

第四章是过渡部分。为使用系统内部功能，先行介绍系统的层次结构。它包括：系统硬件配置基本状况，中断机制和管理，ROM-BIOS的组成和功能，DOS的组成和内存映象，以及DOS中断分类和系统功能（INT 21H）的分类。介绍时尽量不过多涉及硬件知识，并将PC、PC／XT、PC／AT三种微机的系统结构进行对比，以期使读者明确系统资源分布状况和编程使用时的工作环境。

第五～第八章是应用部分。它通过27个汇编语言程序实例来介绍：如何使用BIOS的键盘I／O服务、显示I／O服务、打印I／O服务、日历钟I／O服务、磁盘I／O服务、串行I／O服务等BIOS软件中断功能和BIOS的发声功能；如何使用DOS的字符设备I／O的传统方式和高级方式、FCB式和句柄式的文件管理、目录操作以及取／置日期时间系统功能（INT 21H）；并随之介绍了INT 20H、INT 27H、INT 25H等DOS中断的使用；最后给出了扩充BIOS中断功能的方法和编制驻留程序的实例。每类功能使用举例之前，都用一小节"概述"介绍这类功能实现的硬、软件工作逻辑，重要的数据结构和编程使用时的注意事项。

第九章是扩充部分。它介绍目前广泛使用的FORTRAN、PASCAL、dBASEⅢ、BASIC等几种高级语言与汇编语言的混合编程技术，以期发挥汇编语言可直接访问系统资源的能力和执行速度快的优势，增强用户的高级语言应用程序的功能。本章给出8个实例供读者借鉴。

本书以"贴近读者、简明实用"为宗旨，选材精炼、覆盖面广、实例丰富、论述清晰并便于自学，是PC系列机应用培训班的良好教材，亦可作为高等院校计算机应用专业师生和软件开发人员的参考资料。

作者竭力想奉献给读者一本既有价值、又有特色的书，但由于水平有限，书中不妥乃至错误之处实难避免，敬请广大读者批评指正。

张昆藏

1992.3.

目　录

第一章　中央处理器和指令系统

汇编语言是一种面向机器的程序设计语言。汇编语言程序与其运行的硬件环境有着更为紧密的联系，这是汇编语言区别于高级语言的特点。

本章先从编程的使用角度来介绍 8086／8088 微处理器的内部结构和外部的存访空间结构，再重点介绍寻址方式。寻址方式历来是学习汇编语言的难点，尤其是，8086 微处理器体系采用了存储器分段结构，带来了段缺省和段替换问题。本章后半部分结合实例分类介绍 8086／8088 的指令系统。

为举例方便，本章先行介绍了 DB、DW、DD 等变量定义伪指令和 PROC、ENDP 过程定义伪指令。MASM.EXE 宏汇编程序定义的其它各种伪指令和伪操作符留待第二章系统地介绍。

与 8086／8088 对比，本章也介绍了 80286 微处理器及其增加的指令，目的在于说明实地址模式和保护的虚地址模式的基本概念。

1.1　中央处理器

IBM PC、PC／XT 的中央处理器（CPU）是 Intel 8088，这是一种双列直插式 40 引脚的准 16 位微处理器。"准"的意思是说它的内部是 16 位体系结构，但其外部数据总线仍是 8 位。其指令系统与 8086 完全相同，但 8088 能方便地使用大量的各种 8 位端口器件。

PC／AT 的 CPU 是 Intel 80286，这是一种 68 引脚的正方形四面接线的高性能 16 位微处理器。它有两种工作模式，即实地址模式和受保护的虚地址模式。在实地址模式下，它与 8086／8088 完全兼容。

1.1.1　8088CPU 内部结构

一、指令队列和先行取指方式

8088 内部为 16 位体系结构，有 14 个 16 位寄存器，分为执行部件 EU 和总线接口部件 BIU 两大部分。其结构框图示于图 I.1。

EU 的功能是负责指令的执行，向 BIU 提供数据和地址，管理通用寄存器、指令操作数和标志。除少数管脚（如 S_8）外，EU 和外界是隔绝的。BIU 的功能是负责取指令，读操作数，写存结果等，即和外部总线连接执行各类总线周期。利用 BIU 标有∑的加法器，将段寄存器的值扩大 2^4 倍后，加上段内偏移量，得到 20 位物理地址。因此，直接寻址能力达 1MB。BIU 将由系统存储器取来的指令字节存入指令队列缓冲器，这是一个 4 字节的先进先出存储阵列。队列移出的指令字节由 EU 取走执行。A 总线用于两个部件之间的数据传送。

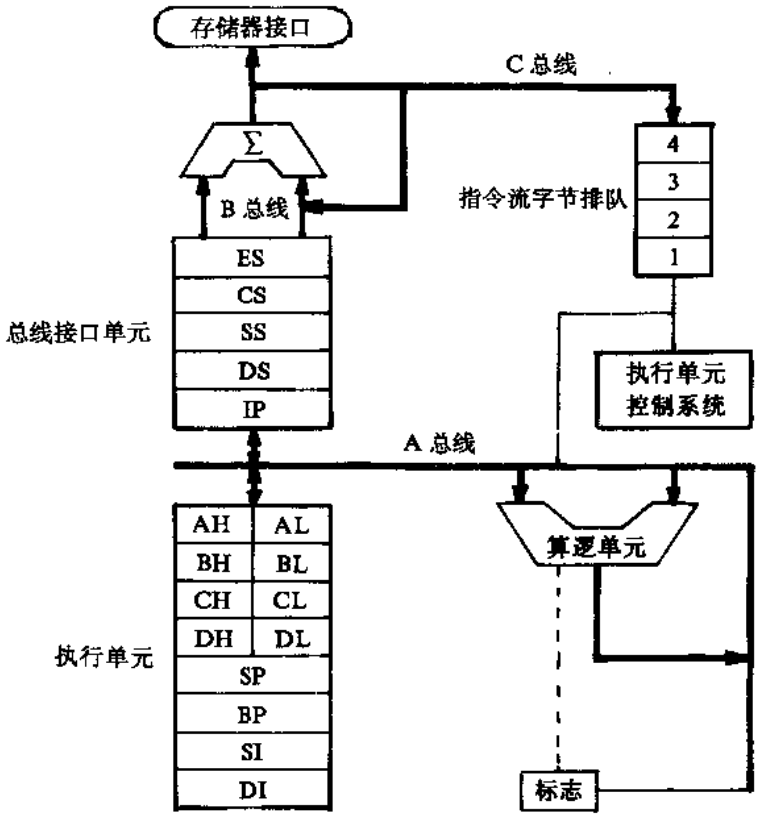

图 1.1　8088CPU 内部结构框图

8088 内部分成可亦对独立运行的这两大部件，是为了在大多数场合下，使取指令和执行指令时间上可以互相覆盖，以提高运行速度。8086／8088 采用先行取指的工作方式——只要在执行当前指令时，CPU 未占用总线和外部交换数据（读或写），它就使用总线从内存取指，除非指令队列缓冲器已满。这就提高了总线的吞吐能力，并且 CPU 一旦执行完一条指令，下一指令就已在 CPU 内部就绪，因而也提高了 CPU 的运行速度。虽然在 EU 遇到转移调用返回类指令时会把 BIU 中的指令队列清除，使先取的指令字节不能使用，但平均而言，分支指令在一个程序中的数目仅占很小比例，故先行取指令的优点仍很明显。为避免 BIU 滥用总线，应合理选择指令队列缓冲器的长度。8086 指令队列长为 6 个字节，并在有 2 字节"空"时才取指；8088 的指令队列缩短到 4 字节，且只要有 1 字节"空"时就取指，这是 8086 和 8088 在内部结构上的差异。

二、内部寄存器

CPU 内部共有 14 个 16 位寄存器，如图 1.2 所示。按其功能可将这些寄存器分为如下 4 组：

1. 数据寄存器

4 个 16 位的数据寄存器 AX、BX、CX、DX，其中每个又可分为两个 8 位数据寄存器。它们用来在 CPU 内部暂存 16 位或 8 位的数据，并有累加器运算能力。但在具体使用上，Intel 还是规定了它们的分工。

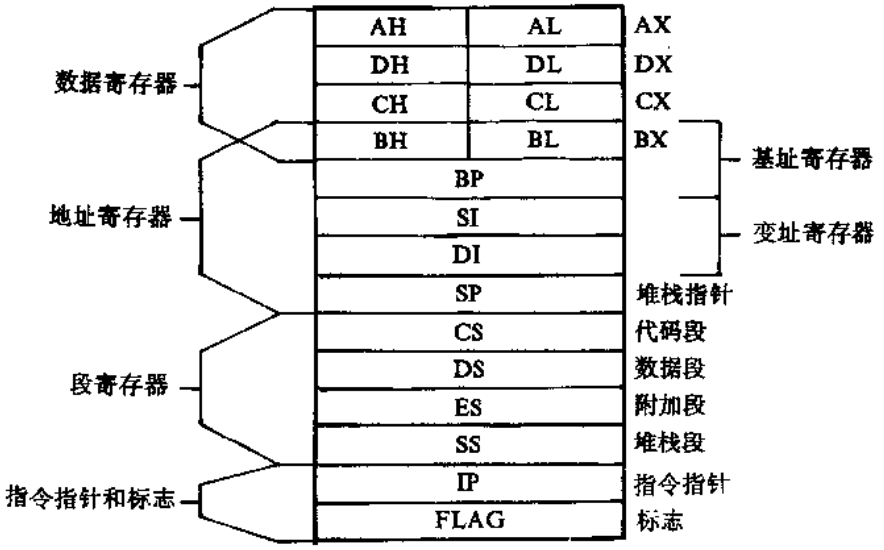

图 1.2　8088 CPU 内部寄存器

AX 寄存器是一个最常用的累加器。对于同样的运算，使用 AX 要比使用其它 3 个数据寄存器能产生更短的指令目标码和更快的执行速度。另外，CPU 与 I／O 端口进行信息交换时，只能使用 AL 或 AX 作为数据寄存器。

DX 寄存器常用来扩展 AX。在进行乘法或除法运算时，DX 用来存放 32 位积或被除数的高 16 位，AX 存放低 16 位。另外，当 CPU 以间址方式访问 I／O 端口时，只能使用 DX 寄存器作为端口地址寄存器。

CX 寄存器常作为迭代控制指令或字符串操作指令的重复次数计数器。CL 寄存器可作为位处理指令的移位计数器。

BX 寄存器可作为一个累加器参加运算，但经常是作为数据段的基址寄存器来使用。这是一个唯一既具有数据寄存器又具有地址寄存器双重功能的寄存器。

2. 段寄存器

8088CPU 对 1MB 存储器的访问采用了分段结构。每个段的最大长度为 64KB，段的起始地址总为 XXXX0h（即可被 16 整除，亦称具有“节”边界）。任一时刻，8088 可寻址 4 个分段，即代码段、数据段、附加数据段和堆栈段。当前可寻址的 4 个分段的段起始地址除以 16 后（即为 XXXXh）分别被存放在 CS、DS、ES 和 SS 4 个段寄存器中。

3. 地址寄存器

由于每个段的最大长度为 64KB，故一旦段起始地址确定之后，对段内的访问只需要 16 位的偏移量即可。16 位的段内偏移量也称为有效地址。8088 提供了灵活的有效地址构成方式，5 个地址寄存器 BX、BP、SP、SI、DI 用于存放有效地址或有效地址的一个分量。在具体使用时，它们有所分工。

BX、SI、DI 用于数据（DS）分段的段内寻址，DI 还可用于附加数据（ES）分段的段内寻址。BP、SP 用于堆栈（SS）分段的段内寻址，但对堆栈的压入、弹出操作自动引用 SP，即 SP 为当前栈顶位置的段内指针。

4. 指令指针和标志

指令指针 IP 是当前代码（CS）分段的段内指针，它指向下一条将被执行的指令（第 1 个字节）的存放位置。即 CS：IP 一起构成程序计数器 PC。IP 是一个专用的 16 位寄存器，它不同于上述的地址寄存器，除了用转移、过程调用、返回等指令来修改其值外，不存在向它直接赋值的指令。

标志 FLAG 是一个 16 位的寄存器，但 8088 只使用了其中的 9 位。各有效位的意义如表 1.1 所示。因各个标志位可单独使用，故一般无需记住它们的位号（单步标志 TF 例外），只要记住它们的符号即可。

表 1.1　标志位的意义

符 号	位号	名称	标志位为 1 的意义
AF	b_4	辅助进位	运算结果的最低 4 位有向上的进位（加法）或借位（减法）
CF	b_0	进位	运算结果的最高位有向上的进位或借位
OF	b_{11}	溢出	运算结果的长度超出目标"容器"的长度（OF＝SF⊕CF）
SF	b_7	符号	运算结果的最高位为 1，若为有符号数则为负数
PF	b_2	奇偶	运算结果中 1 的个数为偶数
ZF	b_6	零	运算结果为 0
DF	b_{10}	方向	字符串操作时，使源、目的指针（SI、DI）自动减量
IF	b_9	中断允许	使 CPU 可响应外部的可屏中断请求
TF	b_8	单步	使 CPU 进入单步工作方式

9 个标志位中 6 个是状态标志，3 个是控制标志。状态标志反映前面运算结果的状态，常用作条件转移指令的条件，故也称为条件标志。控制标志用于使处理器进入某种工作方式。

计算机开机上电时，机内产生的 RESET 复位信号将迫使 CPU 进入如下的初始状态：

FLAG＝清除
IP＝0000h
CS＝FFFFh
DS＝0000h
ES＝0000h
SS＝0000h
指令队列＝空

复位信号出现过后，处理器将由 CS：IP＝FFFF：0000 处取出第 1 条指令执行，这是已固化的一条转至 ROM-BIOS 初始化程序的转移指令。

1.1.2 CPU 可访空间结构

8086／8088 率先打破微处理器只能访问 64KB 存储空间的限制，可寻址高达 1MB 的存储空间。8086／8088 有 20 根地址线的引脚，为能用 16 位的寄存器形成 20 位的物理地址，而又保持寻址方式的灵活性，8086／8088 采用了将存储空间分段的概念，以段和段内偏移量来指定地址。

一、物理地址与逻辑地址

存储器某个位置的 20 位二进制（5 位十六进制）的地址码称为物理地址，它是用来直接驱动 CPU 地址线的。以"段:段偏移量"形式表达的地址称为逻辑地址，它要经过换算之后才得到相应的物理地址。同一物理地址可有多种逻辑地址表达式。

逻辑地址到物理地址的转换是由 CPU 自动完成的。CPU 中 BIU 部分标有∑的加法器总是把指定的某个段寄存器的值扩大 2^4 倍后得到的 20 位段起始地址，再加上 16 位的段内偏移量，从而得到 20 位物理地址。见图 1.3 所示。

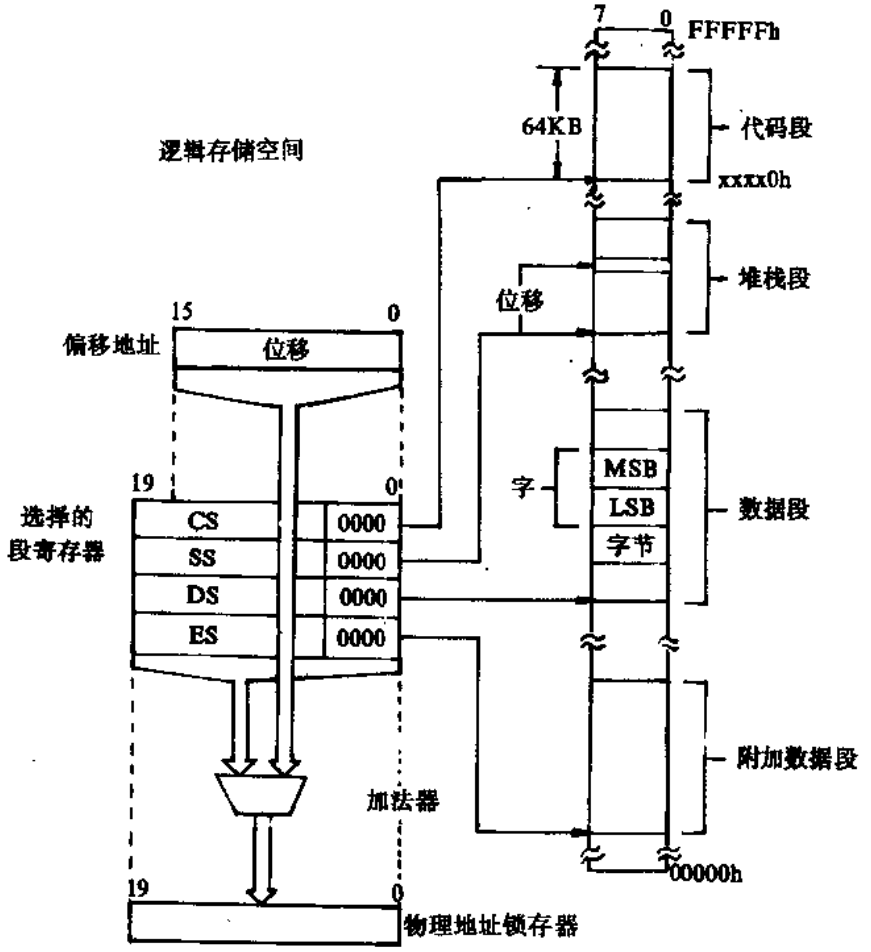

图 1.3　逻辑地址转换为物理地址

例如，代码段寄存器 CS 中的值为 46A8h，指令指针 IP 中的值为 1B23h，则换算过程如下：

$$\begin{array}{r} 46A\ 80 \\ +\quad 1B\ 23 \\ \hline 48\ 5A3 \end{array}$$

于是，CPU 将从 485A3h 这一内存位置开始取下一条指令的代码。

二、段缺省和段替换规则

这种分段结构给初学者带来一些不便，主要是段寄存器名一般不出现在指令的机器码和汇编格式中，而是由操作的性质隐含指定的，这就是"缺省"规则；其次，段寄存器和指针、变址寄存器有较为固定的配用关系，即"搭配"规则；最后，若在指令之前加上"CS:"、"DS:"或"SS:"等前缀，以指定的段寄存器替代隐含的段寄存器，也存在"替换"规则，这些规则见表 1.2。

表 1.2　段缺省和替换规则

存储器存取方式	段 缺 省	段 替 换	偏　移
取指令	CS	不可	IP
堆栈操作 (PUSH POP CALL RET 等指令)	SS	不可	SP
数据存取：*BP* 的间址方式除外	DS	ES、CS、SS	有效地址 EA
数据存取：*BP* 的间址寻址方式时	SS	ES、CS、DS	有效地址 EA
字符串处理指令：源	DS	ES、CS、SS	SI
字符串处理指令：目的	ES	不可	DI

例如，"INC WORD PTR[BX]"这条指令意即以 DS 中的内容（$*2^4$）为段起始地址，以 BX 的内容为偏移量，以"段:"偏移量为逻辑地址求其相应的物理地址，将此内存位置上的 1 个字的存储单元的内容增 1("WORK PTR"伪操作符说明存储单元不是字节型，而是由连续两个字节构成的 1 个字)。这里的缺省段（DS 段）并不出现在指令中。

如果仍以 BX 寄存器的内容为偏移量，且要以 ES 中的内容为段地址完成类似上例的功能，则指令应改写为：INC ES:WORD PTR〔BX〕

三、8088 的 I / O 空间

8088 采用"I／O 单独寻址"方式对外部设备的端口进行编址。在这种方式下，所有的端口地址单独编址构成一个 I／O 空间，而不占用存储空间。8088 以 IN、OUT 这类输入输出指令与这些端口交换信息。

若每次信息传送仅为 8 位，则称这样的端口为 8 位端口。也可以与某端口一次传送 16 位信息，但 16 位的端口实际上极少用。

从原理上讲，8088 可访问 64K 个 8 位端口，但 PC、PC／XT、PC／AT 机中的

I／O地址译码线路决定了只使用1K（1024）个8位端口，地址码范围为000h～3FFh。

对I／O端口的访问一般都采用DX寄存器间址方式，即先把端口地址放入DX寄存器中，然后用一条"IN AL，DX"指令完成从端口的输入，或用一条"OUT DX，AL"指令完成向端口的输出。

如果端口地址在000h～0FFh范围内，则也可使用直接寻址方式，即指令中直接给出1个字节长的端口地址码，如"IN AL，端口地址"或"OUT 端口地址，AL"。

1.1.3 8088、8086、80286的不同

前已说明，8088是一个准16位的微处理器，8086是一个标准的16位微处理器，80286是一个先进的、高性能的16位微处理器。下面简要介绍它们之间的差异。

一、8088与8086的差异

这两类微处理器在内部结构上基本相同，都为16位的体系结构。唯一的区别是8088指令队列为4字节长，只要1字节"空"即取指；而8086指令队列为6字节长，并在2字节"空"时才取指。

它们的外部可访空间结构完全相同。不同处仅在于8088的数据线为8位，8086的数据线为16位。这就是说，在一个存储器读（或写）总线周期内，8088只能完成对内存1个字节的读或写，为读或写1个字（地址相邻的两个字节），必须要两个总线周期。而8086读或写内存偶地址上的1个字只要1个存储器读或写的总线周期。但是要读写奇地址上的1个字仍要两个总线周期，为便于读写1个字节，8086专有1个引脚BHE（高位字节允许）。

由此可见，除非大量的数据字都位于内存的偶地址上，否则，在相同的时钟频率下，8086的内存存取速度并不比8088快。

IBM PC、PC／XT选用的是8088微处理器，其时钟频率选用4.77MHz（时钟周期为210ns）。不加等待的正常存储器读／写总线周期为840ns；I／O端口读／写总线周期由于已固定加入了一个等待时钟周期，故为1.05μs。

二、80286与8086的差异

8086经80186而发展到80286，因此，80286在结构和性能上比8086有很大的提高，这主要体现在80286有实地址模式和保护的虚地址模式的两种工作模式。保护的虚地址模式主要为满足多用户（例如，几个用户通过局域网共享计算机）、多任务（例如，计算机同时运行多个程序）的需要。

1．80286的工作速度更快

这不仅是因为80286处理器的时钟频率可选为6MHz、8MHz或12MHz，远高出8086的5MHz左右的时钟频率，更在于下面两点：

1）80286内部分为执行单元（EU）、总线单元（BU）、指令单元（IU）以及地址单元（AU），这4个相互独立的部分并行操作大大提高了数据吞吐率。

2）80286的地址线引脚和数据线引脚已完全分开，不再采用8086（或8088）那种地址线低16位（或8位）的引脚也用作数据线引脚的分时复用方式。