

● [美] Robert Jourdain 著

● 陈学谦 李民甫 译

IBM PC 编程指南

电子工业出版社

长城

中国计算机发展公司隆重推出
新一代长城系列微机系统

Great WallTM 系列化 · 新标准 · 高性能 · 低价格



TP312
CIP/1

0520/30 系列微型机
程序设计人员工作手册

IBM PC 编程指南

[美] Robert Jourdain 著

陈学谦 李民甫等译

李冠英 谢建勋校

刘德贵 审阅



电子工业出版社

编者的话

由电子工业出版社、北京通联通信电子联合公司和中国 IBM 微机用户协会协作出版发行这套 IBM 微机系列丛书的出发点，就是“快”。所谓快，一是作者著书、翻译、编写要快，二是出版发行要快。这两个环节紧密相联，缺一不可。我们认为这是科技出版界改革的关键所在。

现在奉献给读者的这套系列工作手册，虽是快中之作，但我们完全保证其应用的质量。限于时间、人力、财力，这套书难能做到尽善尽美，个别不到之处在所难免。编辑部将恭听大家的批评指正。

我们的目标是用有限的代价带给您无限的收益。

内容简介

本书根据美国出版的《Programmer's Problem Solver for IBM PC, XT/AT》一书全文译出。该书在 IBM PC 机 XT/AT 的各种技术资料的基础上，结合汇编语言、DOS 调用和高级语言 BASIC 三种编程方式简洁分析了 150 余项常用的硬件控制任务，讲述了许多 IBM 文献中忽略的细节，给出了编程技巧。全书按系统硬件分为七章：1. 系统资源；2. 定时器与发声；3. 键盘；4. 视频显示器；5. 磁盘驱动器；6. 打印机；7. 输入、输出。

该书编排合理，启发性强，为计算机系统开发提供了得力工具，也为初学者开辟了入门捷径。本书可作为广大 PC 机程序设计人员和计算机用户的工作手册，也可作为大专院校、科研单位教学科研的必备参考用书。

译者的话

近年来，我国的微型计算机事业发展很快。目前，应用于各个领域的 IBM 系列各类微机已有 20 余万台。但微机开发应用水平普遍不高，大量微机并未充分发挥应有的作用，其重要原因之一就是微机编程人员不足。

有关 IBM PC, XT, AT 的技术资料种类繁多，如硬件技术参考手册、DOS 技术参考手册、语言参考手册，等等。若不能熟练运用其中的知识，就很难充分开发微机。

一个经过不断学习实践基本掌握了 IBM PC 的编程人员，编程时仍要不厌其烦地查阅各类手册，即使这样也难免有考虑不周之处，从而大大延缓了调试过程。我们一直盼望有一部包罗万象、简洁深入、查找方便的案头书。《Programmer's problem Solver for the IBM PC, XT/AT》即可满足这一需求。

该书集当今 IBM 微机参考书籍之大成，结合三种编程方式（汇编语言编程、操作系统调用及高级语言编程）简洁精辟地分析了 150 余项常用的硬件控制任务，讲述了许多被 IBM 文献忽略的细节，并用一定的篇幅探讨了编程技巧。它以标准 PC 机为基准，讨论了 PCjr, XT 及 AT 型微机各自的特点。该书编排合理、启发性强，既可协助计算机专业人员省时除错，也为初学者开辟了一条入门捷径。

参加本书翻译工作的有：杜桓、黄杰、汪洋、匡、蒋禾光、李民甫、李冠英、谢建勋和陈学谦。我们的翻译工作

从一开始就得到了高庆狮、刘德贵、石国华、叶蜚声等前辈的热情支持；刘璐、华薇、赵北化、卢志奋、秦人华、孙月湘、张淑亚等同志做了大量工作；Norman Easley, Tim Weston 等美国朋友给予至诚的帮助。在此，谨向他们表示衷心的感谢。本书得以顺利出版是与通联通信电子联合公司的大力资助分不开的，特此致谢。

由于我们的水平有限，译文难免有不妥之处，殷切期望诸位同行批评指正。

谨以此书献给他和所有为祖国的昌盛而献身的人们！

陈学谦

1988年4月

序 言

计算机程序设计人员是当今杰出的革新者。遗憾的是，在他们最出色的成果中，有些新技术开发却十分耗时。调试程序所花的时间数十倍于编程时间并非罕见，由于编程人员开始考虑欠周导致程序运行时需反复起动亦屡见不鲜。然而，很少有人论及程序设计中耗时但可能是最需要熟练掌握的是了解机器的有关情况。耗费几个小时来确定一件简单的实据，对一个新程序人员来说是常事，有时令人烦躁得真想把参考书撕烂。

一个普通的程序人员，一夜忙下来被 CRT 弄得眼花缭乱了，打印记录堆积如山。硬件技术参考手册、DOS 技术参考手册、语言参考手册、某些芯片技术资料、打印机和键盘的硬件手册以及成摞的计算机书籍扔得遍地皆是。这些参考资料含有大量信息，要获得这些信息，需要掌握复杂的规则，耗费大量时间。

许多人不具有过目不忘的本领（经常使用计算机反而会使人降低记忆力），所以，这些书确实必不可少。而且，读过的东西还要不时地翻看。初次查找时，也许花了一小时什么资料也没查到。一旦找到，还需经过许多步骤才能从面向初学者那些冗长的说明中筛选出所需要的资料；倘若这本手册是用外语写的，还得用半个下午去翻译，如此等等。可见程序设计人员迫切需要一本大而全的书，其内容尽可能包罗万象，但又不含与编程无关的内容，完全按中等程度编写，

而又囊括了 IBM 的微型机系列，编排合理，查找方便。可这样的书上哪去找呢？

为此目的，作者本人汇集了多方面资料编辑成这本参考书——教科书，以满足那些渴望能编制出色的程序，但又缺少时间（或舍不得花许多钱去买其他手册和书籍）的人之需要。本书在编排上有两个特色：一个是按硬件种类分章，再按硬件性能分节，每节再分条目，一个条目讨论一个特定的编程任务。例如视频这章中有一节讨论光标，本节所设条目将说明怎样定光标、改光标形状、打开及关闭光标等等。另一个是每项讨论都分成四个部分（有的少些），开头一、二段讲述基础知识，然后按高级语言编程、中级的 BIOS 及 DOS 中断编程、初级的直接对微处理器辅助芯片编程等几方面讨论所提出的问题。每章各节开头都用了一、二页讲述了解本节内容所需的一些背景知识。这些概述旨在复习，初次学习的人也不妨一读。

高级编程介绍了用高级语言编写的任务。尽管所用概念完全适用于 PASCAL 或 C 语言，但实例却是用 BASIC 语言编写的。之所以选用 BASIC，一方面它是计算机领域内广泛使用的语言，每个 IBM 微型机用户都配有 BASIC；另一方面，Microsoft 公司的 BASIC 在应用 IBM 硬件方面，是其他程序语言与之不能相比的。即便是 BASIC 语言的初学者，也都能用它来讨论问题。为扩展 BASIC 的功用，本书还给出一些机器语言子程序。在附录中讲了如何把这些子程序编入其它程序的方法。利用这些子程序可以方便地做许多事情，例如键盘再编程、为单色显示卡增加分页功能等等。

中级编程讲述了如何使用操作系统提供的中断功能来完成编程任务。中断是功能很强的小例程，能完成各类计算机

的艰巨工作，比如移动光标、读磁盘目录等。中断是汇编语言的中坚，中级编程中的举例都是用汇编语言写成的。目前，越来越多的高级语言的编译器也能访问中断，使得精明的编程人员能完成高级语言本身不能胜任的工作，如读绝对盘扇区。因此，人们对中级编程的资料越来越感兴趣。本书只讨论了 PC-DOS (MS-DOS)，若要为 CP/M-86 或 UCSDp 系统编程，仍需另找资料。

初级编程中的举例说明了特定任务是怎样在芯片一级上执行的。所有 IBM 微型机系列的结构基本相同，这是因为它们都采用相同的 Intel 系列集成电路。实际上可以用任何语言（包括 BASIC）通过输入/输出端口访问芯片。本书讨论了所有和编程人员关系密切的芯片，诸如定时器芯片、外围接口芯片、中断控制器芯片、CRT 控制器、软盘驱动器以及管理串/并行口的芯片。尽管 IBM 不鼓励编程者在初级水平上编程（因为这种程序在未来的机器上不能运行），但人们不断发现，机器的某些功能用其他方法还实现不了。

并非所有的任务都用这三种编程方法进行讨论，因为有些任务用 BASIC 语言完成不了，有些功能操作系统又不支持。另外有些工作用初级编程很繁琐（如磁盘中的许多操作），加上 DOS 的编写者已经做了这些工作，并且完成得很好，所以对于这些内容没有必要再对其进行讨论。但大多数情况都采用三种编程方法来说明。通过三种方法的比较，可以看到高级语言是如何调用中断的，进而了解中断又是怎样在计算机的心脏——芯片上工作的。

本书看上去也许会让只熟悉 BASIC 或 PASCAL 等高级语言的人望而生畏。这是因为讲述中、初级编程用的汇编语言，看上去象罗塞塔石碑的碑文那样神秘难懂。但对学习汇

编语言的人来说，本书确实是一个理想的伙伴。切不要以为对不懂或不打算学习汇编语言的人，本书就只有三分之一的可用之处，至少有些编译器，如 Turbo Pascal 和 Lattice C 能建立并应用在中级编程中讲述的操作系统功能。实际上，许多初级编程的过程可用高级语言完成。附录 D 备有简介，可充分理解汇编语言实例的执行情况。即使从不使用汇编语言的人，通过了解它的执行过程，也会对高级语言执行的过程及出错的原因有更深刻的认识。

书中的每个小条目几乎都有其程序示例。大多数只是短短几行，有些仅仅是复杂程序中的一小部分，只有少数是可执行的完整程序。书中只给出所需程序的片段以供参考，而不是让好似完美的程序充斥全书，也并非要让所有示例都成为典范。的确，许多程序有意写得不太完善，目的是让读者自己去进一步钻研。举例的目的主要不是提供程序模块大全，而是指点思路，促使读者去思考。倘若愿意，也可将示例转化为实际程序操作的开端，并按需要扩展成特殊的形式。所有示例都经过测试，可以避免因长时间编程而头昏脑胀时所易犯的那些愚蠢错误。

当然，这本书比散文要难读些，但我尽可能少用专业术语，并在书后附上计算机基本词汇表。除某些高深的专业内容外，实际上几乎所有 IBM 文献中有关编程的知识都已编入本书。包罗万象固然不错，但那样本书就将长达千页，而且还会使人只见树木不见森林。对于真正特殊的编程要求，如软盘控制器的扩展编程或 AT 键盘的再编程，还需查阅 IBM 技术参考手册或芯片厂家的性能表。不过，百分之九十九的程序都无需更多的 IBM 硬件资料就可以编写。本书对同一问题给出了不同的解法，并对它们的优劣作了比较。此

外，本书还提供了 ASCII 代码、指令执行时间等各种常用表格。因此，仅用这一本书就能满足全部编程工作的需要。

本书还讲述了许多 IBM 文献所忽略的细节问题，如哪些控制代码是由哪些屏幕输出例程转化的，各类磁盘操作是怎样格式化文件的。有些内容讲述了怎样完成通用编程任务。这些任务原本不是硬件的专职，但却应用了很多硬件的特性，如实时操作和水平卷轴。本书还用一定篇幅讲述了编程技巧。这些技巧虽非出自权威人士之手，但却能解决编程人员所遇到的难题。现在的情形却是：每个编程人员通常都要不止一次地自己弄清这些概念。这种“信息时代的知识分子要象没有纸莎草纸时那样花费大量时间重新发明轮子”的情况应该改变。

本书也介绍了各类 IBM 机器间的差异。以标准 PC 为基准，当 PCjr, XT 或 AT 与之有不同之处时则加以说明。有一点要说明的是，本书没有谈到 AT 机及 DOS 3.0 中涉及多用户系统的特点，这些特点将有专著论述。除了个别注明的例程外，多数程序是针对标准 PC 编制的。若不另加说明，它们可在任何 IBM 微机系列支持下正常运行。还有一点很重要，本书读者须使用 PC-DOS 2.1 或更新的版本及相应的高级 BASIC 版本。

如果说本书对读者有所裨益，主要在于提供了不胜枚举的实据，但愿它们都是正确的。书中还有几百个程序示例，同样也希望完美地奉献给读者，敬请读者检查验证。但本书是在大量资料基础上经多次编辑和修订而成的。由于时间仓促，书中错误在所难免。需要商榷之处，请按下列地址与作者联系（地址：C/O Brady Co, Simon & Schuster, General Reference Group, 1230 Avenue of the Americas, New York,

NY10020)。作者将根据 IBM 最新技术成就修订再版，以使本书更好地为广大程序工作者服务。

Robert Jourdain

* 编者注：罗塞塔石碑是古埃及石碑，碑文用埃及和希腊两种语言和三种文字体系——象形文字、通俗文字（埃及象形文字的草写体）和希腊文字——雕刻而成，解读极为困难。

本书使用的数字说明

使用汇编语言的编程人员，对本书中出现的数字及地址表示方式不会感到陌生。但不少使用高级语言的编程人员却可能不熟悉编址系统和非十进制数字，因此初读时或许有些困惑。然而，本书通俗易懂，读者能很快地熟悉这方面的传统规定。作为一个合格的编程人员，应该认真掌握这些知识。本书后的附录 A 讨论了二进制和十六进制数，及如何用后者表示存储器地址，附录 B 则更全面地讲解了二进制数，以及怎样将其用于处理位操作。编程人员一定要了解下述约定内容：

1. 所有的数目字，除非其后写有 H（表示十六进制数）或 B（表示二进制数），都是指十进制数。当二进制数值显然是表示数字的组合格式时，其后写的 B 可以省略。

2. 另一个例外是形式为 $\times \times \times \times : \times \times \times \times$ 的八位数字。这些十六进制数表示存储器地址的段及偏置。

3. 数字位的计数范围为 0 ~ 7 位（或 0 ~ 15 位），其中 0 位是最低有效位（即置位时，第 0 位 = 1，第 7 位 = 129）。

4. 表达式“ASCII 5”表示 ASCII 字符集的第五个字符。它代表一个数值为 5 的单字节，并不表示符号 5 的 ASCII 代码，也不是数值 5 的双字节整数表达式。

5. 括号内的数，如 [2.1.3]，表示对本书其它章节的交互索引。[2.1.3] 表示第二章，第一节，第三小节。

[2.1.0] 表示第二章第一节开始的回顾性讨论。本书中有

数百处这样的交互索引。通过交互索引，读者可找到刚提到的内容，这种做法对初学者很有用。

6. 书中所用程序代码，其形式总是印刷得很明显，读者极易辨别。

目 录

第一章 系统资源	(1)
第一节 确定系统资源状况	(1)
1.1.1 访问 8255 外设接口.....	(1)
1.1.2 找出 IBM 微机的类型.....	(7)
1.1.3 判定 PC-DOS 版本.....	(8)
1.1.4 找出图形适配卡的数目及类型.....	(9)
1.1.5 找出磁盘驱动器的数目及种类.....	(12)
1.1.6 找出外部设备的数目及类型.....	(14)
1.1.7 确定 RAM 量.....	(17)
第二节 中断管理	(22)
1.2.1 8259 中断控制器的编程.....	(23)
1.2.2 允许/禁止特定的硬件中断.....	(25)
1.2.3 编制自用的中断.....	(26)
1.2.4 完善现有中断.....	(30)
第三节 程序管理	(32)
1.3.1 分配/撤销存储器.....	(34)
1.3.2 运行嵌套程序.....	(38)
1.3.3 程序中使用 DOS 用户接口指令.....	(43)
1.3.4 程序执行完后常驻内存.....	(44)
1.3.5 调入并运行程序覆盖.....	(48)
1.3.6 将程序由·EXE 型转换为·COM 型.....	(52)
第二章 定时器与发声	(58)

第一节	定时器的置数和读数.....	(58)
2.1.1	8253/8254 定时器编程.....	(59)
2.1.2	置/读时间.....	(64)
2.1.3	置/读日期.....	(66)
2.1.4	置/读实时时钟.....	(68)
2.1.5	延时操作编程.....	(71)
2.1.6	定时操作编程.....	(73)
2.1.7	实时控制操作编程.....	(76)
2.1.8	用定时器芯片产生随机数.....	(81)
第二节	发声.....	(83)
2.2.1	76496 发声器的编程方法 (仅对 PCjr)	(84)
2.2.2	演奏.....	(86)
2.2.3	与其它操作同时进行的演奏.....	(89)
2.2.4	报警.....	(92)
2.2.5	演奏音字符串.....	(94)
2.2.6	在进行其它操作的同时演奏音字符串.....	(99)
2.2.7	产生滑音音调.....	(105)
2.2.8	产生音响效果.....	(107)
2.2.9	同时发声.....	(110)
第三章	键盘.....	(112)
第一节	键盘监控.....	(112)
3.1.1	清除键盘缓冲区.....	(114)
3.1.2	检查缓冲区中的键入字符.....	(117)
3.1.3	等待键入字符但不在屏幕上显示.....	(119)
3.1.4	等待键入字符并把它送向屏幕.....	(122)
3.1.5	立即获取字符.....	(125)
3.1.6	获取键入字符串.....	(126)
3.1.7	检查/设置双态键和组合键的状态.....	(129)
3.1.8	编写通用键盘输入例程.....	(132)

3.1.9	重编键盘中断.....	(137)
第二节	存取特殊键.....	(143)
3.2.1	Backspace, Enter, Escape 和 Tab 键的使用.....	(144)
3.2.2	组合键的使用: Shift 键, Ctrl 键和 Alt 键.....	(145)
3.2.3	双态键的使用: Numlock, CapsLock, Ins 和 ScrollLock	(146)
3.2.4	数字副键盘和光标键的使用.....	(147)
3.2.5	功能键的使用.....	(149)
3.2.6	重编单个键.....	(150)
3.2.7	对单个键定义键盘宏功能.....	(153)
3.2.8	建立 Ctrl - Break 例程.....	(155)
3.2.9	PrtSc 键的重定义.....	(157)
第三节	各种键盘编码及应用.....	(159)
3.3.1	各种键的使用.....	(160)
3.3.2	扫描码.....	(161)
3.3.3	ASCII 码.....	(161)
3.3.4	框图编码.....	(164)
3.3.5	扩充码.....	(165)
第四章	视频显示.....	(166)
第一节	视频显示控制.....	(166)
4.1.1	6845 视频控制器编程.....	(168)
4.1.2	设置/检验屏幕显示模式.....	(172)
4.1.3	设置字符属性/颜色.....	(179)
4.1.4	设置屏幕边界颜色.....	(191)
4.1.5	清除全部/部分屏幕内容.....	(193)
4.1.6	视频适配器之间的转换.....	(196)
第二节	光标控制.....	(199)
4.2.1	在绝对位置处设置光标.....	(200)
4.2.2	设置光标于相关位置.....	(204)