

北京希望电脑公司计算机技术丛书

11270

C语言在工程技术上的应用

(上册)

熊可宜 李 宏 编译
谢德刚 周 华



海洋出版社

北京希望电脑公司计算机技术丛书

C 语言在工程技术上的应用

(上册)

熊可宜 李 宏

编译

谢德刚 周 华

海 洋 出 版 社

内 容 简 介

本书通过大量实例清晰地介绍了用 C 语言解决工程问题的方法、特点,内容丰富,实用性强。

全书共分三篇,第一篇介绍 C 语言在电子工程中的应用,包括 C 语言在直流电路、交流电路、晶体管电路及数字系统等方面的应用;第二篇介绍 C 语言在数学分析中的应用,包括 C 语言在矩阵、向量、非线性函数求最佳解、微分方程等方面的应用;第三篇是科技人员使用的 C 语言工具库,包括有 LU 分解、B 样条插值、拟合积分、傅里叶变换、微分方程及工程绘图等程序。

读者对象:科研及工程技术人员,计算机专业教学人员。

欲购本书的用户,请直接与北京 8721 信箱联系,邮码 100080,电话:2562329。

(京)新登字 087 号

责任编辑: 闫世尊

C 语言在工程技术上的应用

(上册)

熊可宜 李 宏

编译

谢德刚 周 华

希 望

审校

海洋出版社出版(北京复兴门外大街 1 号)

海洋出版社发行 兰空印刷厂印刷

开本: 789×1092 1/16 印张:21.25 字数 1070 千字

1992 年 2 月第一版 1992 年 2 月第一次印刷

印数: 1—3000

*

ISBN 7-5027-2514-8/TP·75 定价: 29.00 元/套

前 言

市面上有关 C 语言的书籍比比皆是，但是用 C 语言解决实际工程问题的例子却很少。很多人都认为像电子工程、数值分析等方面的问题要用 FORTRAN 语言来解决，其实不然，C 语言不仅能胜任系统程序设计，在工程上应用起来也是游刃有余的。

本书共分三篇，第一篇介绍 C 语言在电子工程中的应用，包括 C 语言在直流电路、交流电路、晶体管电路及数字系统等方面的应用；第二篇介绍 C 语言在数值分析中的应用，包括 C 语言在矩阵、向量、非线性函数最佳求解、微分方程等方面的应用；第三篇介绍工程师及科学家使用的 C 语言工具库，包括用于 LU 分解、B 样条插值、拟合积分、傅立叶变换、微分方程及工程绘图等 C 语言程序。本书适用于工程技术人员使用，也可作为有关专业计算机辅助教学参考。

编译者
1992 年初

目 录

第一篇 C 语言在电子工程中的应用	1
第一章 计算机和程序设计	1
1-1 目标与建议	1
1-2 程序设计	3
1-3 计算机	3
1-4 编程周期	11
1-6 检查程序	15
1-7 编程错误	18
1-8 改进程序	20
1-9 回顾	20
第二章 C 语言基础	21
2-1 介绍和目标	21
2-2 标识符、数值和存储器	21
2-3 一个程序例子	32
2-4 循环	35
2-5 解方程式	41
2-6 选择	41
第三章 C 语言应用于简单问题	55
3-1 介绍及目的	55
3-2 词首和科学表示	55
3-3 一根电线的电阻	63
3-4 温度对电线的影响	66
3-5 欧姆定律及其类似的关系	74
3-6 在循环中使用 YES 和 NO	75
3-7 电阻器的颜色代码	79
3-8 测试你的知识	86
3-9 复习或回顾	93
第四章 C 语言应用于直流电路	95
4-1 引言和目的	95
4-2 等效电阻	95
4-3 串联和并联电路	99
4-4 分压和分流器原理	101
4-5 梯形网路	101
4-6 求解特征的电路	104

4-7	写自己的函数	114
4-8	复习	121
第五章	编写 C 语言函数	123
5-1	引言和目的	123
5-2	函数的返回值	123
5-3	赋值传值给函数	124
5-4	函数与数组	129
5-5	引用传值给函数	132
5-6	一些用来解决电路问题的函数	141
5-7	复习	153
第六章	程序设计的风格	155
6-1	简介和目的	155
6-2	软件工程	155
6-3	复习	177
第七章	分析 DC 电路	178
7-1	简介及目标	178
7-2	分支电流分析	178
7-3	网分析	179
7-4	点分析	190
7-5	叠加法	194
7-6	THEVENIN 及 NORTON 定理	197
7-7	最大传输功率	200
第八章	瞬时电路应用	207
8-1	简介及目标	207
8-2	与电容有关的计算	207
8-3	与电感线圈有关的计算	214
8-4	复习及回顾	215
第九章	C 语言和正弦曲线	217
9-1	简介和目的	217
9-2	正弦曲线的基本知识	217
第十章	C 语言在 AC 电路中的应用	228
10-1	简介和目的	228
10-2	平均功率及功率因子	228
10-3	复数	228
第十一章	共振及三相电路	247
11-1	简介及目的	247
11-2	共振	247
11-3	三相 AC 电路中的功率计算	256
第十二章	C 语言与电子电路	261
12-1	简介及目标	261

12-2	二极管电路	261
12-3	晶体管电路	273
12-4	运算放大器运算	280
第十三章	频率响应和调制	286
13-1	简介及目标	286
13-2	绘制频率响应图(调频图)	286
13-3	分析——放大电路	290
13-4	调制	293
第十四章	数字系统	303
14-1	简介及目标	303
14-2	记数系统	303
14-3	数字系统和代码	314
14-4	数字系统和真值表	315
附录A		323
附录B	与计算机有关的一些知识	326
附录C	Turbo C 环境和编辑器	327
附录D		329
第二篇	C 语言在数值分析中的应用	331
第一章	几个与数值分析有关的简单 C 程序	331
1-1	C 函数和控制结构复习	331
1-2	一个有用的实例:求质数	334
1-3	阶乘的计算:一个递归实例	336
第二章	一些向量和矩阵的软件工具	338
2-1	dotp 的设计	339
2-2	vdotp()	342
2-3	初值设置和扩充赋值运算符	343
2-4	getvecs()	344
2-5	do-while 循环	346
2-6	break 与 continue	347
2-7	printf()	348
2-8	使用文件指针的文件输入/输出	349
2-9	矩阵相加法	352
2-10	程序员定义的数据结构和矩阵类型	352
2-11	getmats()	357
2-12	madd()	358
2-13	高斯消元法的算法	360
2-14	C 程序	362
2-15	eqsolv 的其余部分	367
2-16	误差和数值扰动	368

2-17	程序开发的组织	369
第三章	矩阵与向量数据的交换	373
3-1	利用管道和 I/O 转向方式交换数据	373
3-2	将管道运用于向量和矩阵数据上	374
3-3	输出函数 <code>fputv()</code> 和 <code>fputm()</code>	377
3-4	错误检查	379
3-5	<code>pgetv()</code>	380
3-6	<code>getdoubles()</code>	383
3-7	<code>getrow()</code>	387
3-8	<code>getbuf()</code>	389
3-9	<code>sscanf()</code>	391
3-10	<code>getword()</code>	392
3-11	洋葱原则(Onion Principle)	394
第四章	绘图语言的接口	397
4-1	绘图语言	398
4-2	解释 <code>graphit</code> 主标准函数	400
4-3	<code>sizeof</code>	406
4-4	窗口—视口和色表的数据存储	408
4-5	割取算法	412
4-6	窗口—视口变换算法	414
4-7	<code>dowind()</code> 和 <code>doview()</code>	415
4-8	<code>doline()</code> 和 <code>doto()</code>	418
4-9	<code>clip()</code>	421
4-10	<code>wvtrans()</code>	426
4-11	<code>grcolor()</code>	427
4-12	实例	430
4-13	在 C 中调用其它程序设计语言	433
第五章	非线性函数最佳解的搜索法	437
5-1	问题独立的软件的设计	438
5-2	求最佳解软件的整体设计	441
5-3	算法的特定数据的局部化	443
5-4	对分搜索算法	444
5-5	梯度搜索算法(The Gradient Search Algorithm)	446
5-6	向量软件工具的 <code>main()</code>	447
5-7	<code>getargs()</code>	450
5-8	<code>vpselect()</code>	451
5-9	<code>switch</code>	454
5-10	<code>vpcheck()</code>	455
5-11	<code>vusage()</code>	457
5-12	梯度搜索算法的 <code>vmethod()</code>	458

5-13	nextx()	463
5-14	对分搜索算法的 smethod()	464
5-15	fdichot.c 和 fgrad.c	467
5-16	实例: 物质需求预测	473
5-17	fgrad 的调用关系图(Call Graph)	476
第六章 解微分方程		477
6-1	积分计算法则所需要的资料流	479
6-2	参数处理及函数的指针	479
6-3	主程序 main()	481
6-4	getargs()	484
6-5	euler 算法	487
6-6	euler()	489
6-7	Runge-Kutta 算法	491
6-8	rkut()	494
6-9	错误的产生	498
6-10	实例	498
6-11	例子: 沙漠消长的模型	503
第三篇 工程师及科学家使用的 C 语言工具库		510
第一章	LU 分解	510
第二章	特征值和辨别分析	537
第三章	奇异值分解: 有效	605
第四章	Newton-Raphson 和相关方法	630
第五章	复数运算 Muller & Jenkins-Traub 方法	648
第六章	B 样条插值法	676
第七章	拟合积分法	695
第八章	傅立叶变换	703
第九章	微分方程组	723
第十章	微分方程刚性系统	744
第十一章	工程绘图	757

第一篇 C 语言在电子工程中的应用

第一章 计算机和程序设计

1-1 目标与建议：

本书的目的在于帮助你学习计算机和程序设计方面的知识，使你在学习 EET（电子技术）和 CCET（计算机工程技术）的过程中，足以胜任简单的编程任务以解决你遇到的问题。这种技能在你的职业生涯中将很有价值，因为许多工程技术人员的工作涉及到计算机。其工作内容很多都与软件相关，许多一开始就对硬件和电子线路感兴趣的人，后来也越来越对编程表现出兴趣。

计算机程序可分作两大类：系统程序和应用程序。系统程序包括你闻所未闻的内容，它们使得计算机接受我们的操纵。系统程序包括文本编辑器、执行系统、输入/输出表、编译程序、解释、汇编程序等。应用程序包括许多你已听说的内容：游戏程序、文字处理程序、银行管理系统、商店信用卡系统、所得税管理程序。控制程序是工程技术人员编写的用于控制工程系统的程序——从较普及的微波炉、电视机、录像机到较新型的项目如汽车防滑制倒器或飞机导航系统。

一个电气系统

作为一个使用程序控制工程系统的例子，请看微波炉：微波炉有一套控制微波加热器加热时间和温度的电子线路。这套线路的核心是一个微处理器——一个可编程电子器件。器件的可编程就意味着制造厂商只需简单地改变微处理器执行的程序就可以改变炉具的工作方式。这样就不必总为下一年的新型号产品专门开发一套新的电子线路了。

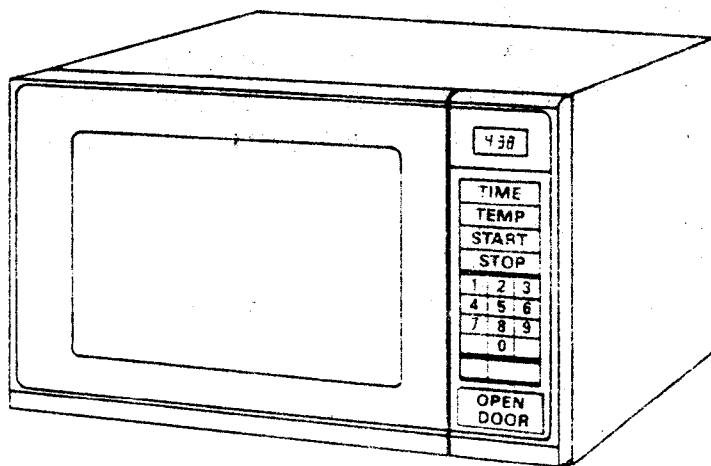


图 1-1

我们虽然不知道微波炉程序的细节，但仍可以决定让程序控制微处理器干什么。由于微波炉配有键盘，程序必须包含告知微处理器怎样从键盘获取信息的指令。同时，微波炉只有一个小型显示器，于是程序就必须包括指出显示什么内容和怎样显示这些内容的指令。程序还必须指出 MPU 怎样在使用者按下“start”键后开始计时。简单地说，程序指示 MPU 怎样控制硬件——键盘、显示器、计时器、加热器等等。在学习编制这样的程序之前，你将先了解 MPU 的硬件基础知识和一种典型的 MPU 指令。

既然许多电气工业中都要用到 MPU，在本篇中，你将学习怎样编制一般程序。稍后，你将学习 MPU 的硬件结构，然后是学习编写对其进行控制的程序。

我们将从编写对电路有用的应用程序开始。在稍后的学习中，随着你所学习的电子方面的知识越来越多，你必将越来越多地学习编写控制程序。但你不大可能成为一名专业程序员。如果你想成为一名专业程序员的话，你也许该专攻计算机技术，而不是工程技术。我的任务是向你提供一些工具，使你能够编写用以解决 CET/EET 方面问题的程序。

C 语言

我选择了 C 语言作为本书的程序设计语言，原因是：

1. 根据美国国家标准局(ANSI)公布的一项规定，用 C 语言编写的程序可在任意型号的机器上使用（也就是说：这些程序是可移植的）。

2. C 语言具有所谓“高级语言”的许多特点。这些特点使得编程者对计算机结构知之不多的情况下仍能编写工作程序。

3. C 语言又具有所谓“低级语言”的特点。这些特点使得编程者在必要时可以利用计算机硬件的特性。这种特点在我们书写控制程序时非常重要。

我们使用的典型的 C 语言是由 Borland 公司推出的 Turbo C 2.0 版本。虽然各种 C 语言都符合 1989 年 ANSI C 的标准，但我仍选用 Turbo C。因为你会发现 Turbo C 的“集成开发环境”非常便于使用。然而，本书中 99% 的程序都不依赖于使用 Turbo C 2.0。Turbo C 可以在任何与 IBM PC 兼容的机器上运行。

目标：

顺利地完成本章学习之后，你将能够：

- * 指出在 PC 机上使用的用于解决在 CET 方面的问题的语言。
- * 确信你自己不会为编程问题所迫而只是成为一个埋头编程的程序员。
- * 说明并解释 PC 机具有一个 CPU（同时也称为 MPU）、易失性主存、非易失性辅存、一副键盘和一个显示器。
- * 解释信息是由程序或数值组成。
- * 解释程序是作为计算机的一个指令序列。
- * 明确 Turbo C 具有一个编译器、一个编辑器和一个连接器。
- * 在程序中使用“include”。
- * 在程序中使用注释（/*...*/）
- * 在程序中使用“main()”。
- * 在程序中使用“puts()”。
- * 了解什么是字符串。

- *了解什么是换行符。
- *了解什么是输出序列。
- *在程序中用到花括号。

注意：不要因为遇到陌生的内容而烦恼，它们既是 C 语言的又是技术方面的内容，本书中将详细解释这些内容。

1-2 程序设计

程序设计是一种使计算机做你所需事情的艺术。作为一个 EET/CET 人员，你可以让计算机为你干许多事情。

例如，在你的电子学初级课程中，你将遇到解线路参数的问题。这些问题包括列出一道多方程式，给一部分未知数的值。对电路进行分析是这种工程技术的一个特征。而设一些数值和解出答案只不过是拥有计算器的初中生的本领。如果你能编程使计算机接收你的方程和赋值，它将为你做一名初中生的工作。使用计算机干这种乏味的工作的好处是它能比人完成得更加精确和迅速。

在计算机应用于 EET/CET 的领域中，还有比线路计算更为简单的方面。你最初学习的知识中包括一张有工程术语前缀和与它对应的十进制权重的表格(例: "micro"表示 10^{-6})。如果你编程打印一张总表的话，在需要时你手头就有一份参考手册。

图 1-2 是一个非常简单的 C 语言程序。你将在本章后半部分见到这个程序的详尽解释。但是如果在更深入地学习之前能看一些例子的话，以后对 C 语言的描述将更容易一些。大胆地猜测一下程序各行的功能，你将惊奇地发现你是何等地正确！

```
/* my first C program! */

#include <stdio.h>

main()
{
    puts("\n\nhello,world!\n");
}
```

图 1-2 一个简单的 C 程序

1-3 计算机

以下是一份被认为是 PC 机的计算机的清单。

1. IBM PC, PC XT, PC AT, PS/2 型机的模式，等等。
2. 工作时与这一系列 IBM 计算机相似的机器，包括 Zenith, Epson, Compaq, Tandy, Toshiba, NEC, AT&T, Hewlett_packard, AST Research 等公司的机器以及许多美国和远东出产的“杂牌”计算机。这一类机器往往被称为兼容机或衍生机，但我在此称它们为 PC 系列机。

3. Apple II, Macintosh 和其类型的 Apple 机。

4. Commodore 64 和 128 型能及 Amiga。

5. Atari 家族。

一台 PC 机通常用一个微处理器作为 CPU（中央处理器——计算机的“大脑”）；有存储器；有一台类似于打字机的键盘；一台视屏显示器；一些“大容量存储器”，同时，这一装置也被称为“辅助存储器”。例如，一个或多个磁盘驱动器；或许机器还配有一台打印机。CPU、存储器和磁盘驱动器通常是安装在机壳内，键盘和显示器在机壳外部。如果键盘和显示器装在一个机壳中，它就被称为视屏显示终端。

CPU

CPU 的功能是执行程序的指令。如果一条指令告诉 CPU 从内存的某一指定位置（称为“地址”）取数并且将这个数与另一个从内存另一指定地址取出的数相加，CPU 将先访问第一个地址，将地址中的数据拷贝到暂存器中，然后访问第二个地址，拷贝其中的数据到另一暂存器，再将两暂存器中的数相加后，将结果写回暂存器中。CPU 拥有一个算术——逻辑部件（ALU）用以实现各种运算。

所有计算机都有 CPU，在某些机型上，CPU 是一个微处理器，因此对于这些计算机，我们既可以将它的大脑称为 CPU，又可称为 MPU。微处理器是电子集成技术所创造的奇迹。在一小块硅片上制造出许多相互连接的简单电子线路并将它们联接成一个复杂的设备，这就构成了微处理器。

内存

内存是 PC 机中存放信息的地方。它就像你做实验时使用的笔记本。笔记本上记录着关于实验操作的步骤，还空出了记录测量数据的位置。计算机中信息的内容分为两大类：程序和数据。数据是被 CPU 所处理的信息。例如：前缀“micro”和数字 10^{-6} 是两段信息，它们包含在关于工程名词前缀以及与之关联的十进制的表中，处理这一类信息最多不过是把它们送到屏幕上显示。对于一些数据则要求对其进行计算。例如：你上个周末的加班工时和表示你每小时应得加班费的数据，就可以被当作数据项。对它们的处理包括将二者相乘以求出你的总加班费。正如我讲过，一个程序就是一个 CPU 执行的指令序列。但是这种解释还不够。一条程序就是一条完整的指令序列，它告诉 CPU 怎样具体地完成特定的任务。与 MPU 一样，一台 PC 机的主存储器是在一块硅片上集成的电子线路，它通常被称为 RAM（注意：RAM 曾经是一个正规的缩写词，但是今天它仅仅用来特指主存储器）。

大容量存储器

大容量存储装置之所以必需是因为主存——和 CPU——同处于机壳内的存储器是非稳态的。当多们关掉机器的电源时，RAM 中的所有信息都会丢失。当我们再次打开电源时，不确定的值——我们称之为无用信息——将会被写入所有的存储地址之中，于是我们存放在存储器中的值就会彻底地丢失。基于这种情形，我们还需要稳态的辅助存储器来保存我们工作的结果。这种辅助存储器多用磁盘来担当。从使用录音机的经验中你可以知道：磁介质是相对稳定的。当记录器在磁盘上磁化了一些点后，这些点在被抹去或重新磁化前保持原有的磁性。计算机磁盘是在具有较大存储量的同时价格较低廉的存储装置，因此我们能够大量的信息安全地存放在磁盘上。我用于编写本书的计算机有两台软磁盘驱动器和两台硬磁盘驱动器。磁盘驱动器和磁盘之间的关系就好比是磁带机和磁带的关系。磁盘的存放信息的介质，

而驱动器说是从磁盘上读／写信息的设备。

在 PC 机上，磁盘驱动器被表示为从“A”开始的字母。按照惯例“A”和“B”是软盘驱动器，而“C”则是硬盘驱动器。你所使用的机器也许与此有不同。我使用的一台软磁盘驱动器（简称软驱）能使用直径为 5.25 寸的软盘；这种软盘可以容纳 360,000(360KB——

“K”代表“1024”)个“字符”的信息；一个字符是一个字母或二进制数或标点符号。图 1-3 表示的是一个 5.25 英寸软盘。正如盒式录音带一样，软磁盘使用一种非常柔软的塑料（由此得名“软磁盘”），表面涂以磁性物质。这部分是磁盘的主体（有时被称为“介质”）。它往往被永久性地置于一个保护壳中，壳往往是一黑色的厚纸壳。绝不要试图将塑料盘体从盒中取出！盘盒上有一个开口，使得驱动器的读／写头能在磁盘转动时接触盘面。绝不要从盘盒上的任何开口触摸磁盘！你购买的所有的软盘都带有氏套。当磁盘不在驱动器中时，将磁盘放在套子里。盘套保护介质不受手指、铅笔等东西的接触。

另一个驱动器使用 3.5 英寸软盘。一块 3.5 英寸盘可以存储两倍于 360k5.25 英寸软盘的信息。图 1-4 表示的是一张 3.5 英寸软磁盘。它的介质装在一个硬塑盒中，有一块滑动盖板保护着读／写口。这种软磁盘不需要专门的盘套。台果用硬盘的话，每张硬盘能够存放 24,000,000 个字符。硬盘和驱动器通常是一体化的，也就是说，介质不能从驱动器中取出。这使得硬盘驱动器的设计者不仅能使系统保存比相同大小的软盘更多的信息，同时还能以比软驱快得多的速度读／写。然而，这也使得硬盘系统对震动很敏感，所以不要拍打计算机或把书往工作台上扔。

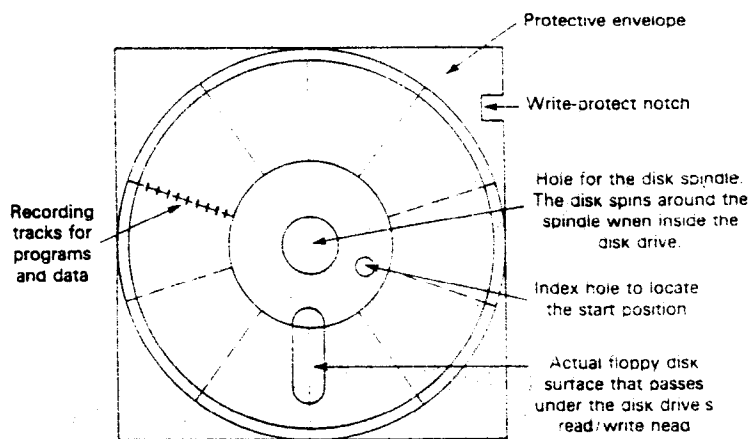


图 1-3 5.25 英寸软盘

在使用 5.25 英寸盘时，先将它从盘套中取出，把盘放入驱动器中，关上驱动器门（3.5 英寸盘没有盘套，开关驱动器门也是自动的）。这时你就可以键入命令从主存将信息调入磁盘，或从磁盘向主存写入信息，或者还可以用一个程序自动传送数据。当信息传输结束后，把软盘从驱动器中取出放回盘套中保存起来。不要让磁盘受潮、受热、受坚硬东西的擦刮、受弯折、受压、受磁场影响等等。

软盘在第一次使用之前，它必须被初始化，或称“格式化”。软盘与盒式录音带或录像带不同，你无法在不进行初始化之前先在盘上写入信息。在格式化时磁盘盘面将被分成有

确定数目的同心“磁道”（就像靶子上的圆环），并且将每个磁道划分成一定数目的“扇区”（就像馅饼的切片）。接着，格式化程序在每个扇区中写入标号（还有其一些信息），以使得计算机能够检索记录在磁盘上的信息。计算机在检索时，可以从一张表上找到所需信息的位置（即：磁道数和扇区数）。接着计算机驱动磁盘读/写头移动到信息所在磁道，当指定扇区从读写头通过时，驱动器就读入所需信息。格式化也会将原有磁盘信息抹去，这样，你原来在盘中存的东西就会丢失。所以，除非是对刚买的新盘，否则你对软盘格式化之事必须小心，因为说不定盘上记录着你的实验数据！

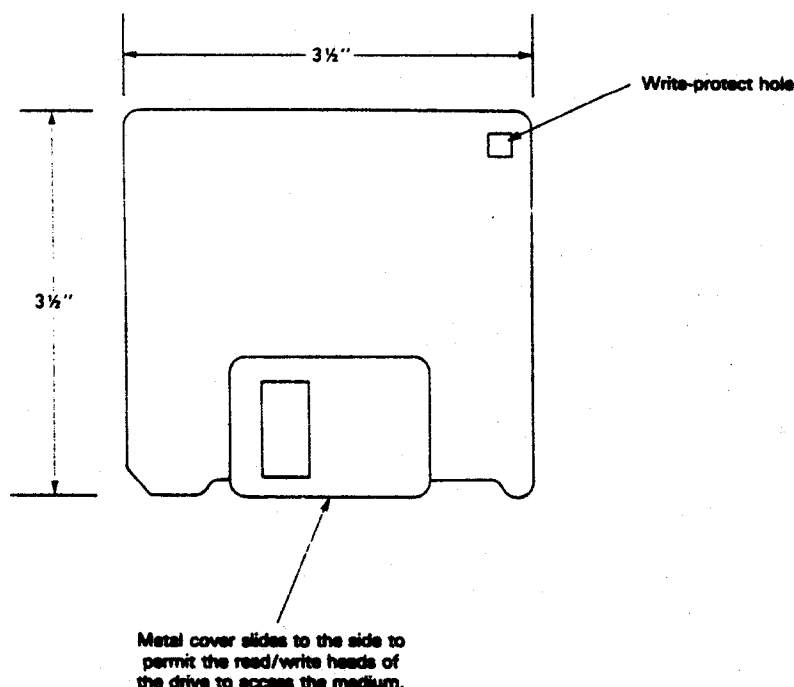


图 1-4 3.5 英寸软盘

软盘最大是直径 8 英寸的。磁性材料制造技术的发展使 5.25 英寸盘应运而生，再后来又出现了 3.5 寸盘。令人惊奇的是：软盘在物理尺寸上越来越小的同时，它的容量却增大了，过去的 8 英寸软盘只能存储 250,000 个字符。表 1-1 记录着软盘的容量“M”表示“K²或 1,048,576”。

硬盘驱动器内装着的“唱片”大小从约 3.5 英寸还小到 14 英寸不等。然而，硬盘盘片大都无法从驱动器中取出，所以也许你永远看不到它。

键盘

键盘用于向机器输入信息和发出指令。我们称通过键盘的输入为键入，比如说“键入 A”就表示“敲击代表着‘A’的键”。我称这一动作为“敲击”是因为很多键盘都是自动重复输入——如果你压住这个键足够长的话——一般是 0.5——0.7 秒。你可以将使用 Turbo 的编辑器将你键入的信息放入内存。内存中存放你键入信息的部份称为编辑缓冲区或文本缓冲区。这些信息可能是 C 语言程序或是程序所需的数据。当你键入一个命令告诉编辑器将信息

从区拷贝到磁盘的一个“文件”上。一个文件出就是一些相关信息的集合——程序的步骤、数据，一张你朋友的姓名和住址清单等等。一张纸上记录的文件放在文件夹中，就正如计算机文件放在磁盘上一样。

表 1-1 软盘的发展

Diameter (in.)	Number of characters	
8	250,000	(250K)
5.25	360,000	(360K)
5.25	1,200,000	(1.2M)
3.5	720,000	(720K)
3.5	1,440,000	(1.44M)

你可以键入你看到的任何一个键，以及一些键的组合键。例如，当你敲击“A”你输入的是“a”。想要输入“A”你必须压下 shift 并且敲击“A”键然后放开两键。确切地说，有三个“shift”类的键：shift,Ctil 和 Alt。“A”是 shift-A 的缩写，“^A”是 Ctr-A 的缩写(Alt-A 无缩写)。

显示器

显示器让你知道现在正在进行什么过程。例如，许多你键入的命令会在屏幕上显示出来；这称之为“回应”。当一个程序给出结果——例如，一张关于前缀和与之相应的十进制权重的示格，或是一个电路计算的结果——将在屏幕上显示出——如果你的程序中包含有告诉 CPU 将结果输出到屏幕上的指令的话。如果你的计算机系统带有一台打印机，你可以获得表格或计算结果的一份“硬拷贝”甚至上一份源文件的拷贝（这被称为表），方法是给出命令让计算机将结果输出到打印机。我们计算机室每三台计算机配有一台打印机，通过手动开关可以连接不同的机器。为了获得硬拷贝，学生们在确认同伴中没有人要使用打印机之后，将开关接通他的机器然后使用“TYPE”命令把程序拷贝到打印机。你们的机房也许和我们是一样的，也许每台计算机都配有一台打印机，上机指导书将告诉你怎样在上机时获得硬拷贝。

操作系统

一台计算机仅仅是硬件——一个包括电子线路（如：CPU、内存）、磁性材料（如：磁盘）和电机（如：驱动器、打印机）的子系统——除非它还带有一个必须的程序，否则，计算机什么事也干不了。对 PC 机（或其通用计算机）这种软件称为操作系统。IBM FC 的操作系统称为 PC_DOS，改进的系统是 MS——DOS(Microsoft Disk Operating System)。如果你有一台 PC 机，你就应该配有 PC_DOS，如果你有一台改进型的机器，你就应该配备 MS_DOS。我的意图是将 PC_DOS 和 MS_DOS 相等同。我将二者都称为“DOS”。操作系统控制着计算机所有子系统的运行。

当你打开计算机电源时，机器通过读取硬盘上的几个特殊程序实现自动初始化过程——称之为自举。没有硬盘的机器通常用一张专用软盘——称之为系统盘——实现自举，系统盘往往在开机前就被置于“A”驱动器中。你的上机指导书有关于你使用的操作系统的细节内

容。“自举”一词来源于“用自己引导功能把自己拉起来”，一个看来不可能成功的尝试。这种说法是因为看上去计算机似乎是能不用程序的帮助开始运行的。实际上，自举依靠的是安装在机内电子器件中的程序。这个电子器件被称为自举 ROM。ROM 是一种电子存储块，因此它是硬件，而程序是软件。这种固化在芯片中形成的结构体就称为固件。

在自举过程的结尾，计算机（实际上是 OS）显示一种被称为提示符的符号以表示它准备接收你的命令。提示符包括当前驱动器号——通常在软盘系统下是 A，在硬盘系统下是 C。当前驱动器是你可直接对置在其中的磁盘进行操作的那个驱动器。一台双软驱的 PC 机通常用 A、B 标识两个驱动器，A 驱动器在上边，B 在下边，或 A 左 B 右。你的上机指导书将告诉你有关你所用计算机的问题。

以下是一些标提示的例子：

```
A:> C:> C:\TURBOC> C:\TURBOC\INCLUDE>
```

第一个例子是表示你当前驱动器是 A，而第二个例子是表示 C 是你当前驱动器。剩下的两个例子表示了用“\”分隔的目录和子目录。目录和子目录就好比是办公室中的一个文件柜中的一个抽屉中的一个栏目。小栏目可能包括纸张。可以把文件看作是磁盘驱动器，把抽屉看作是目录，把大栏目当作子目录，纸张和小栏目看作是文件。例如：“C:\TURBOC”表示在文件柜“C:”中的一个栏目“TURBOC”。“C:\TURBOC\INCLUDE”表示在文件柜“C:\”中的一个抽屉“TURBOC”中有一个栏目“INCLUDE”。在办公室中，你也许想到文件或在栏目“TURBOC”中找到更一些小栏目。在计算机中，文件就是程序或数据。图 1-5 表示 DOS 文件系统的分层次序列结构。（见下面图）