



● 一九九一年 ● 总期第73期

4

電子

744

與 電腦



• ELECTRONICS AND COMPUTERS •

电子工业出版社向广大的计算机爱好者推荐

中华学习机实用大全系列丛书

BASIC与LOGO语言

定价 4.40元

汉字处理与数据库技术

定价 4.60元

操作系统

定价 2.20元

FORTRAN与Pascal语言

定价 3.80元

汇编语言程序设计

定价 2.50元

游戏与绘图

定价 2.50元

硬件维修与经验技巧

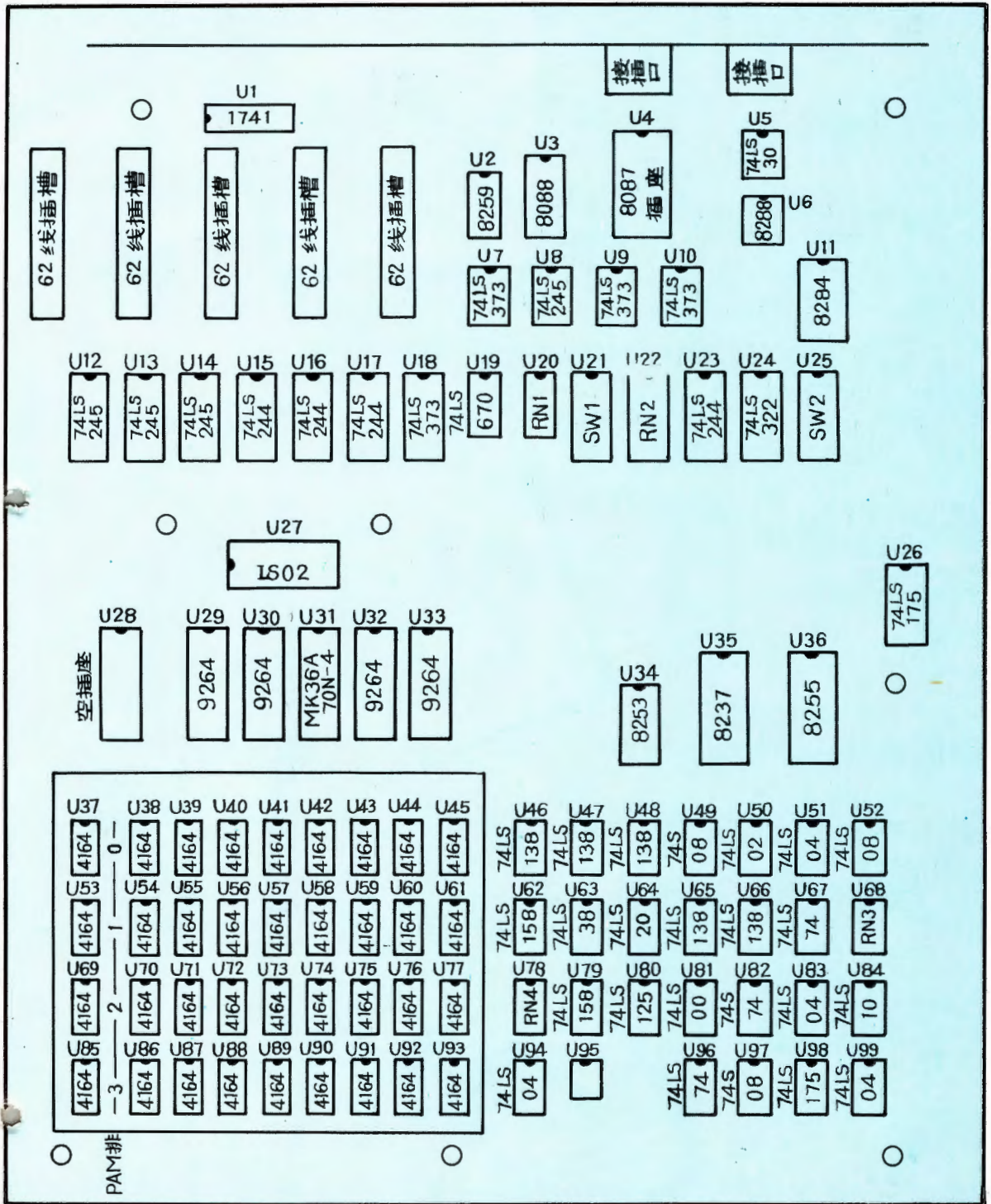
未出版

有需要者请与北京173信箱 发行部邮购联系，外加15%的邮资费。



香港 余健康 供稿

管脚号	当主板上电后，用逻辑笔测试每个芯片的管脚信号，对照此查寻表，即可找出故障																																									
芯片号	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40		
U1	P	L	L	P	P	P	H	O			P	L	L	L	P	P	P	P	L	H		H	H	L	L	H	P	P	H													
U2	P	P	H	P	P	P	P	P	P		P	L	L	L	P	P	P	P	L	H		H	H	L	L	H	P	P	H													
U3	L	P	P	P	P	P	P	P	P		P	P	P	P	P	P	P	L	L	P		L	P	H	P	P	P	P	P	H		H	P	L	H	P	P	P	P	P	H	
U4	40脚DIP插座																																									
U5	H	H	H	P	P	P	L	P	O	O		P	P	O	H																											
U6	L	P	P	P	P	P	P	P	P	L		P	P	H	P	P	P	P	P	P	H																					
U7	P	P	P	P	P	P	P	P	P	L		P	P	P	P	P	P	P	P	P	H																					
U8	P	P	P	P	P	P	P	P	P	L		P	P	P	P	P	P	P	P	P	H																					
U9	P	P	P	P	P	P	P	P	P	L		P	P	P	P	P	P	P	P	P	H																					
U10	P	P	P	P	P	H	O	O	H	L		P	H	O	O	H	P	P	P	P	H																					
U11	L	P	P	P	P	L	H	P	L	L		H	P	L	O	L	P	P	H																							
U12	H	P	P	P	P	P	P	P	P	L		P	P	P	P	P	P	P	P	P	H																					
U13	P	P	P	P	P	P	P	P	P	L		P	P	P	P	P	P	P	P	P	H																					
U14	P	H	P	P	P	H	H	H	H	L		H	H	H	H	P	H	P	H	L	H																					
U15	L	P	P	P	P	P	P	P	P	L		P	P	P	P	P	P	P	P	L	H																					
U16	P	P	P	P	P	P	P	P	P	L		P	P	P	P	P	P	P	P	P	H																					
U17	P	P	P	P	P	P	P	P	P	L		P	P	P	P	P	P	P	P	P	H																					
U18	P	P	P	P	P	P	P	P	P	L		P	P	P	P	P	P	P	P	P	H																					
U19	P	P	P	H	H	P	P	L	P	P		P	H	P	P	P	H																									
U20	—RN1—电阻网络																																									
U21	—SW1—磁盘驱动器选择开关																																									
U22	—RN2—电阻网络																																									
U23	H	L	L	H	L	L	L																																			





一九九一年

总期第73期

電子與電腦

目

· 综述 ·

90年代的电脑技术和采购策略(上) 殷广济(2)

· PC 用户 ·

用随机文件建立数据新探 纪京松(5)

英文 Wordstar 文件的标准 ASCII 化 赵毅(7)

BASIC 程序 P 保护在文件加密中的应用 何骏(7)

在 dBASE III 中实现多维数组功能 石供泰(8)

为 DOS 增加定义功能键外部命令 张红(9)

设置 MS-DOS 的功能键 何管略(11)

编译 BASIC 运行文件 CCDOS 不相容的

解决方法 刘亚兵(12)

硬盘的加锁和解锁 张文(12)

键控图形处理程序 徐维科(13)

彩色的高分辨率图形 潘麟元(14)

回溯法及其在编程中的应用 陈磊(15)

· 学习机之友 ·

中华学习机使用技巧 任义(16)

实用改良的 NEWCODE—POKES 程序 黄影昌(17)

巧解“八皇后”问题 柳杨(17)

关于 APPLE 和 CEC-I 机数计算误差的讨论

——对《APPLE 和—I 设计者的疏忽》一文看法 刘峰(18)

——APPLE 缺陷的解决办法 陈盛施(18)

——数值计算中存在问题的解决办法 朱洪波(18)

无数据磁盘格式化程序 姜宏岩(19)

跟踪机器语言程序的运行 田俊成(20)

录

局部清屏程序 程华(20)

扩展 BLOAD 命令的功能 蒋阳(21)

从“重赏”谈起 卢保堂 卢玉堂(21)

在 APPLE SOFT 下使用小汇编 稽克威(22)

工资票面张数计算程序的简化 孙承灿(22)

· C 语言初阶 ·

第四讲 C 运算符 李文兵(23)

· 学用单片机 ·

智力竞赛抢答装置 宋济华(27)

单片机电子音乐编程实践 张培仁 张新宇(30)

· 学装微电脑 ·

数字游标长尺的制作 易齐干(34)

· 初级程序员级水平考试辅导讲座 ·

微机数据录入操作回答 刘英 张茂芝(40)

· 维修经验谈 ·

软盘驱动器机械故障的分析与维修 魏学则(44)

· 更换集成块一法 吴广元(44)

APPLE-II 机内存故障维修方法 腾达(45)

巧修 CEC-I 受损键盘 黄文贵(45)

· 新书与软件 ·

VIRUSCAN 病毒处理软件包功能简介 朱禹(46)

Microsoft Windows 3.0 窗口软件简介 (46)

最新版本 PC TOOLS V6.0 (47)

· 封二、封三 · IBM PC 主板芯片故障查寻表

IBM PC 主板光器件排列图

机械电子工业部电子工业出版社主办

编辑、出版：《电子与电脑》编辑部

（北京 173 信箱 邮政编码：100036）

印刷：北京三二〇九厂

国内总发行：北京市邮政局

国内统一刊号：CN11-2199

邮发代号：2-888

国外代号：M924

出版日期：每月 23 日

主编：王惠民 副主编：王昌铭

责任编辑：徐云鹏

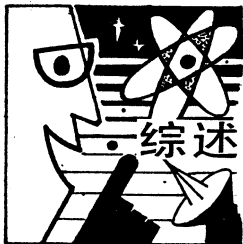
订购处：全国各地邮电局

国外总发行：中国国际图书贸易总公司

（北京 399 信箱 邮政编码 100044）

广告经营许可证：京海工商广字 147 号

定价：0.95 元



90 年代的电脑技术和采购策略(上)

北方电脑公司 殷广济

自从 1981 年美国 IBM 公司推出 PC 机以来,至今仅有十年历史。目前,PC 机市场上品种五彩缤纷,到处皆是。从主流机种的中心,即称作 CPU 的芯片发展历史来看也还不到 20 年的历史。曾记得 1984 年 8 月当新的 IBM PC/AT 首次推出时,IBM 的代表声称:这部新的电脑功能太强了,不能只将它视作单用户个人电脑,同时,他们向在场的人士保证,PC/AT 的主要用途将是连接 2—3 台终端机的多用户系统的核心。当时,各方面的专家也赞成这种评估,并认为没有人真正需要象 PC/AT 般强大的计算能力。

然而,从那时候起,事情发生了巨大变化,拥有 512KB 内存,6MHz 的 PC/AT 不再是个人电脑市场上的宠儿了。几年后的今天,PC/AT 已进入其生命周期的衰退期,大有取代 PC/XT 作为电脑入门机器的趋势,32 位电脑成为主流的时代已经到来。

采用 80386,80486 作 CPU 的电脑系统(以下称 386,486 系统)和采用 80286 作 CPU 的 286 系统最显著的差异在于 386,486 系统 CPU 是 32 位的,目前 386CPU 的时钟速率有 16MHz,20MHz,25MHz,33MHz 四种,486CPU 的时钟速率有 25MHz,33MHz 两种,速率越快的处理机,指令执行速度亦快得多,从而使 386,486 系统的整体功能比 286 系统为佳。

386 系统的运算速度达 5MIPS,486 系统的运算速度达 20MIPS,基本上接近过去大型机的水平。采用 33MHz 80486 作 CPU 的个人电脑创造了速度记录,快,确实是快。但从市场和电脑系统设计角度来看,仍然会因 BIOS 及硬件兼容性,数学协处理器的运用,总线、扩充能力的不同而影响系统功能。要把 90 年代电脑技术变成采购策略是困难的。

好比一个健康的人,除头脑、四肢强壮之外,身体各器官的功能也需要正常。同样,若 386,486 系统只靠 CPU 运算速度快并不一定有用,存储器及输入输出(I/O)设备存取速度的配合也相当重要,若存储器及 I/O 装置的速度太慢,CPU 就会时常被迫停留在静候的状态(Wait States),整体系统功能无法发挥。

造成 CPU 访问存储器时有等候现象发生的原因是通常使用 DRAM 作为主存储器,这种主存储器每隔一段时间就需要重新充电一次,否则电荷漏光,原本应为“1”的信息变为“0”,影响输出结果的正确性。所以,当 CPU 要访问存储器时,通常会对存储器充电一次,而充电时间越长,就会造成 CPU 的等待现象。

在 1985 年,最好的处理器是 80286,时钟频率仅有 6/8MHz,8MHz 的 80286 访问存储器时需要在 125ns 时间增量内完成,在 250ns,375ns 内完成,相对

于零等待就是 1 个,2 个等待状态。在处理器和存储器子系统间等待状态数被当成恒量系统性能的尺度。

到了 1986 年,80386—16MHz CPU 问世,恒量等待状态的时间增量成了 62.5ns,再使用传统的 DRAM,系统就不可能达到真正的零等待,一些厂家如果按 8MHz 的 80286 作为恒量的标准,为了提高自己产品的性能,自称(错误的)他们的 386 系统运行在零等待状态,实际上是 1/2 个等待状态,他们担心的是他们的机器不是运行在零等待状态,比 8MHz 的 AT 零等待状态的机器运行还慢,系统功能也随之降低。

解决办法之一是将系统的 DRAM 全部换为速度较快(<80ns)的 DRAM,当然,系统的价格会稍贵些。

第二种解决办法是大家看到的 Cache memory 方式,就是以一部分 SRAM (Static Random Access Memory)来取代 DRAM。其原理是按照统计,程序在大部分时间只是访问较少范围内的内存地址,采用 SRAM,就是把经常存取的数据放在数量虽少,但速度很快的 SRAM 上(这种存储装置不需要充电,在同等条件下,访问速度比 DRAM 快 2—3 倍),不常用的数据放在 DRAM 上,一样可以增加系统的存取速度,由于 SRAM 比 DRAM 贵得多(在相同价格前提下,DRAM 能提供的存储容量差不多是 SRAM 的 4 倍),所以通常只能以一部分 SRAM 来代替 DRAM,而不是全部采用 SRAM,一般采用 32KB 或 64KB 的 SRAM,这样,访问数据的命中率可达到 90%以上,而系统造价仍较便宜。

还有些优化 DRAM 存储系统结构的方法,如交互式 and 猝发转换结构等,来改进系统性能。

所谓交互的意思是在不同的 DRAM 体的存储地址交替进行预充电,使得对一个 DRAM 体进行访问时,对另一个 DRAM 体预充电,导致系统存储器循环时间等于 DRAM 访问时间,可以不计预充电时间(差不多占 DRAM 访问时间的一半),和采用 SRAM 差不多。

猝发转换是把 DRAM 体上的数据移到高速缓存 SRAM 上提高性能方法。

因此,在采购电脑系统时应注意 DRAM 的速度指标及结构,是否与 CPU 相匹配。

在 PC/XT、AT 时代,兼容性并不是一个很严重的问题,或许说根本不是问题,因为蓝色巨人“IBM”提供了标准版本,兼容机厂商都以此版本为依据设计产品,只要支付 IBM 公司总产值的 1% 的许可证费即可,而不必支出大量的开发费,进行低成本的生产,低价在市场出售,1988 年兼容机的产量达 570 万台,远

远超过 IBM 公司 180 万台的产量,大大威胁 IBM 公司市场。

为此,IBM 公司采用新策略,开发了新的微型计算机系统 PS/2,独自の操作系统 OS/2,IBM386 系统使用微通道(micro channel)结构,首先使用 3.5"软盘驱动器。同时,把生产许可证收费提到 5%,兼容机公司走与 IBM 兼容道路,利益风险太大,于是,COMPAQ,AST,Award,HP,NEC 等发表了与 IBM 公司 MCA 结构(micro channel architecture)相抗衡的扩展工业标准(Extended, industry standard architecture),1989 年 7 月,Intel 公司按 EISA 标准设计的 80386 芯片推出,同年 11 月推出 EISA 微机,90 年 2 月,EISA 标准的 80486 微机问世。所以,采用 Intel80x86 系列芯片的微机有三种结构,即 PC 及兼容机,MCA,EISA。在缺乏标准版本且厂商数目众多的情况下,用户对这些产品的相容性是值得注意的。

相容性的问题可以分为两个层次:BIOS(Basic Input/output System)相容性和硬件相容性。BIOS 的相容性是较容易做到,一般用户写的程序通常由键盘或磁盘驱动器存取信息,这通常是由系统提供的子程序来完成的,如果各系统提供的 BIOS 都能接受相同的变量,就可以说是 BIOS 兼容,对目前的 386 系统,若能执行 DOS 下的软件,则 BIOS 相容性是没有问题的。

硬件的相容性相对地困难,有人写程序习惯不通过 BIOS,而直接到某个地址访问信息,以争取时间,这时两个硬件系统兼容,则要每个地址扮演完全一样功能,在不涉嫌仿制的前提下是相当困难的。

目前,各种 386 系统之间相容性的问题就在于扩展卡的相容性,除了 IBM 的 PS/2-80 外,所有 386 系统均有 8 位,16 位,32 位扩展槽。毫无疑问,PS/2-80 的扩展卡不可用于其它系统上,因为总线的结构不一样,而其它系统的 8 位,16 位扩展槽,其目的在于让用户使用以前 XT,AT 的扩展卡,以降低成本。但 80386 的速度比 80286,8088 快得多,所以在使用这些扩展卡时,系统的速度必须放慢,也就是系统整体功能降低,有些厂家把系统速度限制减慢到某一限度,如 12MHz,此时某些扩展卡就不能在该系统上使用。

缺乏标准的 32 位扩展槽是另一个问题,每个厂家产品的 32 位扩展槽的接脚可能都不同,用户在购买前,务必弄清可以接哪些 32 位扩展卡,或者一次买齐,否则,买了卡以后而无法扩充系统功能,徒劳着急。

在购买电脑之前,应该审视整个系统,如果采用速度较慢的硬盘系统和显示系统,即使你用世界上最快的 CPU,你的系统也会象一条蜗牛。

硬盘驱动器一开始便与计算机事业同步发展。在 PC/XT 的时代,硬盘驱动器的存取时间大约为 85ms,AT 的硬盘驱动器存取速度大约为 40ms,由于个人电脑技术不断推进,磁盘驱动器技术也随之向前进步,为了配合 80386 的高速度,目前市场上硬盘驱动器的存取速度大约是 28ms,有的产品甚至提高到 16ms。目

前,一般个人电脑的硬盘驱动器容量已进入 20MB 至 40MB 的范围,且大量采用 3.5 英寸的半高硬盘驱动器,平均的标准读写时间为 20 多 ms。由于网络技术及桌面印刷系统等发展,市场对大容量硬盘驱动器的需求增加,相继出现多种新产品,包括 300MB,600MB,1.2GB 的大容量硬盘驱动器。随着膝上型,笔记本型电脑的问世和普及,新一代 2.5 英寸 20MB 薄型(一英寸厚)的硬盘驱动器乘时而起。

3.5 英寸软盘驱动器较 5.25 英寸软盘驱动器设计较好,生产技术日渐成熟,成本下降,价格便宜,适应个人电脑微型化趋势并受膝上型电脑的普及所帮助,3.5 英寸 1.44MB 软磁盘驱动器亦将取代传统的 5.25 英寸 1.2MB 软盘驱动器的地位,2.5 英寸小型软盘驱动器也已面世多时。

现在,已有一些 4MB 的磁盘,一些制造商正在开发特殊的技术以适应 10M 容量及更高容量产品的需求。

由于光盘存储器记录密度高,存储容量大,随机存取,盘片可更换,信息保存寿命长,环境要求低,工作稳定可靠,是一个有前途的新兴外存储器设备,在市场上光盘存储器大约有三种类型:只读型,一次写入型和可抹型,容量均为 1000/3000MB。可改写的光盘系统也已面市,光磁盘容量虽然大大超过所有的硬磁盘,但缺点是存取时间稍慢,约为 40MS,传输速率也小,约为 2MB/S,但其成本价格却低于硬磁盘,且盘片可以更换,特别适用于图象处理,印刷系统,办公室自动化的文件存档系统和 CAD 系统,但由于技术及价格原因,短期内未必能普及。

除了存储介质以外,显示屏也是用户必须考虑的重要因素。即使你的电脑拥有再高的处理能力,如果不能将信息准确地传达给使用它的人便毫无意义。因此,对电脑画面的显示要求具有更高的清晰度。IBM 公司的 PS/2 产品未问世之前,彩色显示器都以 EGA 为标准,现在有了解析度更高,色彩更丰富的 VGA,PVGA 显示器,通过提高点距离密度和高对比度实现了高度清晰的显示,无论多复杂的图表,三维几何图形以及笔划繁杂的汉字均可色彩鲜艳地显示出来,甚至微妙差别亦清楚表示,分辨率有 640×480,800×600,1024×768,1280×1024 甚至更高,这取决于要你的电脑如何生动活泼,选择适当的显示器。

在 80386 族谱中,还有个弟弟 80386SX,也是 Intel 公司 32 位产品之一。内部结构与 80386 相同,采用 100 接脚方形扁平塑封,体积比 80286、80386 都小,能耗也小,可以采用表面安装技术。芯片的外部数据总线为 16 位,能使电脑厂家以不太贵的 80286 元件来设计主板,可以 16 位机的价格来推销 32 位系统。

在许多公司计划开发 386SX 机时,对其前途莫衷一是。有的视其为闪闪发光的明星,有人说它对 16MHz 的 286 和 386 电脑市场冲击不大,有人认为正是令最终用户难以选择的另一类产品。

因为标准的 386 是以 32 位递增访问存储器,但 386SX 则以 16 位递增进行访问,因而芯片通过速度较慢。不去和 25MHz、33MHz 的 80386 芯片相比较,那样比也太公平,16MHz 的 386SX 和 16MHz 的 80386 相比,平均慢 23%,你要买真正的 32 位电脑系统,还是用 80386 好。虽然 386SX 和速度相当的,甚至和还要快一点的 286 芯片相比,也没有压倒优势,价格也较贵,但在网络服务器的应用中 386SX 就显得特别重要。如果你要运行 DESQ View 或 Lotus 1-2-3,买

80386SX 比买 80286 系统合适,如果你不想运行全 32 位软件,简单地说只是运行目前 DOS 和 OS/2 下的软件,买较快的 80286,比如 20MHz 或 25MHz 的系统,有更快运行速度与 DOS 或 OS/2 软件兼容,价格相宜。

- 80X86 系列芯片的新一代新星是 80486,它已开始用于个人电脑领域以及以性能为关键的工程工作站和小型计算机市场。(待续)

加快软件商品化进程 推动计算机广泛应用

本刊记者

机电部计算机司委托中国计算机软件与技术服务总公司(简称:中软总公司,英文缩写 CS&S)举办“91 全国第三届计算机软件交流交易会。”会议于 91 年 4 月 4 日~9 日在北京国贸中心举行。来自全国各地的近百家单位近千种应用软件进行展示交流,形成了一定规模的软件商品市场。

所展示的软件大致可分为三类:一类是国家级,部委,大企业事业单位开发的用于国家、部委、城市系统的管理软件,工具软件和专用软件包。二类是许多通用性的软件,如多种财务方面的管理软件,文字处理(照排),以及有关的改进 DOS 汉字处理功能的软件等。三类是引进和汉化的国外先进软件,如 SQL 语言、

PCTOOL V6.0,汉化 window3.0 等。

用著作权法保护软件已成国际潮流,我国的著作权法将于 91 年 6 月实施,软件的出版发行工作势在必行,在这一新形势下,许多大学出版社的软件出版部纷纷在展示会上亮相,如北大出版社软件部;南大出版社软件部;华中理工大学出版社软件部等在会场设置摊位,进行试探性的促销活动,提高自己的知名度。可以预言将来大部分个人和集体开发的优秀软件将会以出版社软件出版物的形式进行推广并得到保护。

注:本文标题为机电部曾培炎副部长为“软交会”的题词。

全国青少年校外计算机活动研讨会

征文通知

由中国科协发起,并纳入 UNESCO IIP 项目总体规划,决定于 1991 年 8 月中旬在福建省厦门市举行《全国青少年校外计算机活动研讨会》。具体日期将在 1991 年 8 月全国青少年信息学(计算机)竞赛期间。该会议由中国计算机学会和中国科协青少年科技中心主办,厦门市承办。现将征集论文事项通知如下:

一、会议主题:计算机活动对青少年教育的影响

二、征文的主要内容:

1. 计算机普及与未来人才;
2. 中小学普及计算机知识的内容、方式探讨;
3. 计算机普及教育对其它中小学课程教育的促进与辨证关系;
4. 从认知心理科学角度探讨计算机普及教育的迁移作用;
5. 青少年校外计算机活动中心的模式与功能分析;

6. 综述性论文。这一类论文要求有具体统计资料、事实例证、学术观点和科学分析。不可以投寄空洞无物或工作总结式文章;

7. 开发先进的计算机教育软件(含示范表演)。

征文要求

文章一般限 5000 字,前面附有 300 字摘要(摘要用中英两种文字),书写在 400 字稿纸上,力求文字语言、标点符号正确通畅,图表规范,标出图题和标题及序号并在文中放在合适的位置上。一式二份于 1991 年 6 月 10 日前寄到:北京师范大学 计算中心 谢文杰同志处

(邮编:100875 电话:(01)2012288-2708)

四、重要日期

1. 截稿日期 1991 年 6 月 10 日
2. 发出录用通知 1991 年 6 月 30 日
3. 会议统一编印论文集。



PC 用户

用随机文件建立数据库新探

纪京松

很多单位大都用 DBASE Ⅲ 建立数据库管理数据。但 DBASE Ⅲ 资源开销大,速度又慢,所以,一些系统完全可以用 BASIC 语言来实现。我所近几年组建了一些大、中型水利自动化管理系统,都是用 BASIC 语言建立随机文件对水情数据进行存储管理的。

水情关心的只是数据的时间、站名,各数据间没有网状联系。所以,我们以时间为基础换算出各数据存取所需的记录号。例如:以一个小时为时段。

$CODE = MONTH * 31 * 24 + DAY * 24 + TIME$

7 月 7 日 8 时的数据记录号便是:

$CODE = 7 * 31 * 24 + 7 * 24 + 8$

按此记录号,便可以存取所需要的水情数据,文件中也无需存储时间数据,可节省硬盘资源。各水文站站名、水库名也无需存储,只要各记录中各字段次序一定下来,就可以按位置找到各水库的水情数据。下面举几个简单的例子说明一下:

(一)存放四个水库的雨量数据:

程序一

```
5 DIM A1$(4),A$(4)
10 OPEN "S" AS #1
20 FIELD #1,10 AS A$(1),10 AS A$(2),
    10 AS A$(3),10 AS A$(4)
30 INPUT "月: ";MONTH
40 INPUT "日: ";DAY
50 INPUT "时: ";TIME
60 INPUT "A 水库雨量: ";A1$(1)
70 INPUT "B 水库雨量: ";A1$(2)
80 INPUT "C 水库雨量: ";A1$(3)
90 INPUT "D 水库雨量: ";A1$(4)
100 CODE=MONTH*31*24+DAY*24+TIME
110 FOR I=1 TO 4;LSET A$(I)=A1$(I);NEXT I
120 PUT #1,CODE
130 CLOSE;END
运行:RUN
```

月:7

日:7

时:8

A 水库雨量:264.33

B 水库雨量:32.1

C 水库雨量:5.00

D 水库雨量:102.51

这样 7 月 7 日 8 时,四个水库的当时雨量数据就存入随机文件"S"中,记录号为 5384。

(二)取出四个水库 7 月 7 日 8 时的雨量数据

程序二

```
5 DIM A1$(4),A$(4)
```

```
10 OPEN "S" AS #1
20 FIELD #1,10 AS A$(1),10 AS A$(2),
    10 AS A$(3),10 AS A$(4)
30 INPUT "月: ";MONTH
40 INPUT "日: ";DAY
50 INPUT "时: ";TIME
60 CODE =MONTH*31*24+DAY*24+TIME
70 GET #1,CODE
80 PRINT "A 水库雨量: ";A1$(1)
90 PRINT "B 水库雨量: ";A1$(2)
100 PRINT "C 水库雨量: ";A1$(3)
110 PRINT "D 水库雨量: ";A1$(4)
120 CLOSE;END
运行:RUN
```

月:7

日:7

时:8

显示:

A 水库雨量:264.33

B 水库雨量:32.1

C 水库雨量:5.00

D 水库雨量:102.51

以上例子只管理了四个水库的数据,每一项定了 10 个字节。如果要管理成百上千个水库的数据,既浪费资源且无法实现,因随机文件一个记录最长只有 32767 字节。因此需修改数据库结构,进一步节约硬盘和内存资源开销,才得以管理上万个站的水情数据。

下面还是以 4 个水库的例子来说明。原来每个记录缓冲区占 40 字节,现在把它压缩成 8 字节。

程序三

```
10 REM 新思维建立数据库
20 DIM A$(4),NAM$(4),A1$(4),A(4)
30 COLOR 15,1;CLS
40 LOCATE 6,9;PRINT "输入日期: 月 日 时"
50 LOCATE 6,20;INPUT MON
60 LOCATE 6,27;INPUT DAY
70 LOCATE 6,34;INPUT TIM
80 FOR I=1 TO 4;READ NAM$(I);NEXT I
90 CLOSE;OPEN "Y.DAT" AS #2
100 FIELD #2,2 AS A$(1),2 AS A$(2),
    2 AS A$(3),2 AS A$(4)
110 FOR I=1 TO 4
120 PRINT NAM$(I);;INPUT A1$(I)
130 NEXT
140 FOR I=1 TO 4
150 A(I)=VAL(A1$(I))*100;A1$(I)=MKI$(A(I))
```

```

160 LSET A$(I)=A1$(I)
170 NEXT I
180 CODE=MON*31*24+DAY*24+TIM
190 PUT #2,CODE
200 CLOSE;OPEN "Y.DAT" AS #2
210 FIELD #2,2 AS A$(1),2 AS A$(2),
      2 AS A$(3),2 AS A$(4)
220 GET #2,CODE
230 FOR I=1 TO 4
240 A(I)=CVI(A$(I))/100;A$(I)=STR$(A(I))
250 LON=LEN(A$(I));LON=LON-1
260 A$(I)=MID$(A$(I),2,LON)
270 PRINT NAM$(I);A$(I)
280 NEXT I
290 CLOSE;END
300 DATA "A 水库雨量:", "B 水库雨量:",
      "C 水库雨量:", "D 水库雨量:"

```

运行时输入

月:7✓

日:7✓

时:8✓

A 水库雨量:264.33✓

B 水库雨量:32.01✓

C 水库雨量:5.00✓

D 水库雨量:102.51✓

雨量单位 mm

保留两位小数

输出:A 水库雨量:264.33

B 水库雨量:32.01

C 水库雨量:5.00

D 水库雨量:102.51

其中,150句:A(I)=VAL(A1\$(I))*100把A1\$(I)转换成相应的2个字节的内部代码字符串。

这样,存入随机文件中的数据,实际上是各雨量数据相对应的2个字节的内部代码,因此可大大地节省机器资源开销,完成大量数据的存储。

240句:将内部字符串还原成原来的字符串,为以后的定位修改,显示再存储提供条件。

250~260:消除符号位,水情数据都为正数。

以上的程序,表明此方法比用DBASE III 建库优越得多,而且编译BASIC速度也快。由此推广开去就能建更大型的数据库,每个记录存取更多的数据。

随机文件一个记录最长为32767字节,这就意味着一个记录最长可以存取16383水情数据。

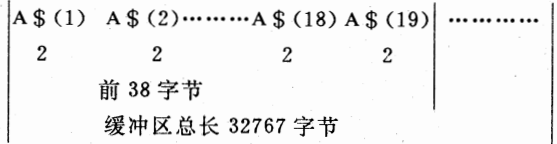
一个很棘手的问题是如何把一个记录缓冲区划分成16383个区段。用一个FIELD语句肯定是无法实现的。因为BASIC每个语句的长度最多不能超过256个字节,其中还包括行号。经过实验一个FIELD最多可划分19个区段。即

```

100 FIELD #1,2 AS A$(1),2 AS A$(2),...
      ...,2 AS A$(18),2 AS A$(19)
110 FIELD #1,2 AS A$(20),2 AS A$(21)...
      ...,2 AS A$(38)

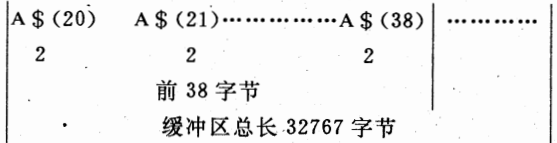
```

如此划分下去不但麻烦,而且也不对。经过多次反复研究试验,发现110句的FIELD语句并不象某些教科书上所写继续从第39个字节开始划分以后的区段,而是又从头开始,如图一、图二说明。



图一

图一是执行100句的结果,前38个字节划分成19个区段,分别为A\$(1),A\$(2).....A\$(19)。



图二

图二是110句的执行后结果,又把前38个字节划分给了A\$(20),A\$(21).....A\$(38)。

因此,要使110句从第39个字节开始划分,则110句2 AS A\$(20)前面必须加一个偏移量,就是把前38个字节划给一个区段A\$。即

```

110 FIELD #1,38 AS A$,2 AS A$(20),.....
      ...,2 AS A$(37)

```

为了方便,我们采用循环语句完成。以下是建立一个A.DAT随机文件,需要存放500个数据的程序。

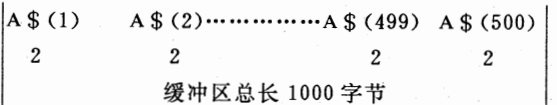
程序三

```

90 CLOSE;OPEN "A.DAT" AS #1 LEN=1000
100 FOR I=1 TO 500
110 FIELD #1,2*(I-1) AS A$,2 AS A$(I)
120 NEXT I

```

110句A\$为偏移量,执行结果如图三。



图三

另外,BASIC有良好的绘图功能和通信功能,系统中的地图绘制等雨量线、水库剖面、各测站的通信都能用高级编译BASIC完成。这样的管理系统是DBASE III 所无法实现的。

如果数据过大,可采用单、双精度,转换成4、8个字节的内部代码时用MKS\$,MKD\$函数。

当然,用户也可以根据自己的实际情况计算记录号,安排数据的存取,建立所需的数据库。

工作中得高昆高级工程师的大力帮助,在此深表感谢。

英文 WordStar 文本文件的标准 ASCII 化

河北师范学院 赵毅

英文 WordStar (4.0 版) 是文字处理中常用的软件。它的功能丰富, 简便易学。由于配备了一个主辞典和一个用户辞典, 我们可以用机器来检查和更正输入文本的拼写错误。最具特点的是与之配套的收有 22 万词条的同义词词典, 机器响应迅速, 对于经常用英文写作的用户来说, 的确是个好帮手。然而, 经它编辑出的文本文件采用的不是标准 ASCII 码, 无法用其它的系统来显示, 重新编辑或打印。笔者谨为新用户介绍解决这一问题的三种常用方法。

(1) 在主菜单 (Opening Menu) 下打 N, 选用非文本方式, 该方式采用的是标准 ASCII 码。输入欲处理文件的文件名, 打回车键。打开文件后, 键入命令 ^B 或 ^QU。处理完后存盘即可。注意此法不会删掉文件中已设定的打印控制命令字符。

(2) 在主菜单下打 P, 进入打印操作状态。在最后一个打印提示语句后键入 ASCII。这样, WordStar 会动处理你的文本文件并以 ASCII.WS 的文件名存盘。你也可以自己规定文件名, 具体方法是在键入 ASCII 后打一个 > 符号, 然后输入文件名。注意该法保留原文件, 但会删除新的文件中的控制字符。

(3) 有些用户需要写作中英文两种文字夹杂的文件。他们通常用中文 WordStar 来输入汉字并排版, 用英文 WordStar 来处理英文。下面为大家介绍笔者用 BASIC 编写的一个小程序, 该程序在进行标准 ASCII 化处理的同时, 能够在屏幕上显示处理结果:

```
10 CLS; LOCATE 5, 20; INPUT "原文件名", S$
20 LOCATE 8, 20; INPUT "新文件名?", T$; CLS
30 OPEN S$ FOR INPUT AS #1; OPEN T$ FOR OUTPUT AS #2
40 IF EOF(1) = -1 THEN 140
50 X$ = INPUT$(1, #1); M = ASC(X$)
60 IF M < 128 THEN 80
70 IF M <= 160 AND M >= 128 THEN X$ = CHR$(M-128)
80 PRINT #2, X$; ; PRINT X$; ; GOTO 40
90 Y$ = INPUT$(1, #1); N = ASC(Y$)
100 IF N < 128 THEN X$ = CHR$(M-128)
110 PRINT #2, X$, Y$; ; PRINT X$, Y$; ; GOTO 40
120 IF N > 160 THEN 110
130 X$ = CHR$(M-128); Y$ = CHR$(N-128); GOTO 110
140 CLOSE #1, #2; CLS
150 END
```

BASIC 程序 P 保护在文件加密中的应用

何骏

BASIC 程序有三种方式保存文件。

1. SAVE "文件名" / — 压缩二进制格式。
2. SAVE "文件名", A / — ASCII 格式。
3. SAVE "文件名", P / — 编码二进制格式。这是一个保护性选择项, 当程序经过 P 保护后, 试图对它发出 LIST, EDIT 命令都会失败。

笔者编了一个程序 KEY20, 利用这一点来对文件进行加密, 其主导思想是把 ASCII 码文件的每一句加上行号及注解符 " ", 使之能在 BASIC 状态下用 LOAD 命令调入。

加密时先运行 FKEY20.BAS, 输入需加密的文件名, 然后返回操作系统状态, 把文件扩展名改成 .BAS, 再进入 BASIC 状态, 用 LOAD 调入, 用第三种方式保存文件, 最后恢复原先的文件扩展名。这样加密后就不能在 EDLIN 或 WS 下进行编辑。

解密时, 在 BASIC 状态下用 LOAD 把加密的文件调入, 先破 P 保护 (注: 有关破 P 保护的方式有几种, 在此从略), 然后用上述方式 2 保存文件, 再运行 FKEY20.BAS, 最后恢复原先文件名即可。

```
10 REM FILE-KEY PROGRAM VERSION 2.0
20 KEY OFF; CLS; DIM A$(1000)
30 PRINT "FILE-KEY PROGRAM VERSION 2.0"
40 PRINT; PRINT "PROGRAMMED BY HEJUN"; PRINT
50 INPUT "PLEASE INPUT FILENAME (EXIT --- ENTER)"; F$
60 IF F$ = "" THEN SYSTEM
70 INPUT "LOCK OR UNLOCK (1 OR 0)"; K
80 IF K <> 1 AND K <> 0 THEN 70
90 OPEN "I", #1, F$; I = 0
100 IF EOF(1) THEN CLOSE #1; GOTO 120
110 I = I + 1; LINE INPUT #1, A$(I); GOTO 100
120 OPEN "O", #1, F$
130 IF K = 0 GOTO 180
140 FOR J = 1 TO I
150 C$ = STR$(J); L = LEN(C$); C$ = MID$(C$, 2, L - 1)
160 A$(J) = C$ + " " + A$(J); PRINT #1, A$(J)
170 NEXT J; GOTO 220
180 FOR J = 1 TO I
190 P = INSTR(A$(J), " "); L = LEN(A$(J))
200 A$(J) = RIGHT$(A$(J), L - P); PRINT #1, A$(J)
210 NEXT J
220 CLOSE #1; ERASE A$; GOTO 20
```


在 dBASE Ⅲ 中实现多维数组功能

内蒙哲里木盟饲料厂 石供泰

由于 dBASE Ⅲ 没有数组语句,使得许多计算问题难以解决,但是人们巧妙地运用 & 函数,很容易地实现了一维数组,如程序一, A&J 相当于 BASIC 语言中的 A(J) 语句,只要注意循环变量 I 为“N”型,应用 J=STR(I,1) 将其转化为“C”型才能使用 & 函数。

对于 BASIC 语言中的 A(J,N), 则不能简单地用 A&J&N 表示,这样表示在 J、N 小于 10 程序能够正常运行(如程序二)。在 BASIC 语言中的 A(1,11) 或 A(11,1), 也就是在 J=11 或 N=11 时, dBASE Ⅲ 中 A&J&N 都表示为 A111, 分不清是 A(1,11) 还是 A

(11,1), 这就不能正常进行数组计算。

笔者曾想到在 dBASE Ⅲ 中数组 A&J&N 中间加字母 B 的办法即 A&JB&N, 但程序运行后出现了不可认字符, 即 &JB 不可认, 为此笔者用 &X 代替字母 B, 即 A&J&X&N (相当于 A2B3 的形式), 实现了二维数组在 dBASE Ⅲ 中的计算的通用性(如程序三)。

同理用 A&J&X&N&Y&E (其中 &Y 为 C 型, &E 为数值变量) 相当于 A1B2C3G 形式, 代替 BASIC 语言中的三维数组 A(1,2,3) 实现了多维数组在 dBASE Ⅲ 中应用的通用性。

```
程序一:
SET TALK OFF
CLEA
I=0
DO WHILE I<9
I=I+1
J=STR(I,1)
A&J=I*20
ENDDO
I=0
DO WHILE I<9
I=I+1
J=STR(I,1)
? A&J
ENDDO
```

```
程序二:
SET TALK OFF
CLEA
I=0
DO WHILE I<9
I=I+1
M=0
DO WHILE M<9
M=M+1
J=STR(I,1)
N=STR(M,1)
A&J&N=M*20+I*100
ENDD
```

```
ENDD
I=0
DO WHILE I<9
I=I+1
M=0
DO WHILE M<9
M=M+1
J=STR(I,1)
N=STR(M,1)
? A&J&N
ENDD
ENDD
```

```
程序三:
SET TALK OFF
CLEA
I=0
X="B"
DO WHILE I<15
I=I+1
IF I>=10
J=STR(I,2)
ELSE
J=STR(I,1)
ENDIF
M=0
DO WHILE M<15
M=M+1
```

```
IF M>=10
N=STR(M,2)
ELSE
N=STR(M,1)
ENDIF
A&J&X&N=M*20+I*100
ENDD
ENDD
I=0
X="B"
DO WHILE I<15
I=I+1
IF I>=10
J=STR(I,2)
ELSE
J=STR(I,1)
ENDIF
M=0
DO WHILE M<15
M=M+1
IF M>=10
N=STR(M,2)
ELSE
N=STR(M,1)
ENDIF
? A&J&X&N
ENDD
ENDD
```

为 DOS 增加定义功能键外部命令

湖北省公路局科研所 张 红

许多高级程序设计语言,如 BASIC、DBASE III 都提供了对功能键进行定义的功能。这种“软键”功能使用户能根据需要将一些经常使用的关键字或字符串赋予功能键。这样,在要输入这些关键字和字符串时,用户只需按一次功能键就能完成,不必逐个字符输入,从而简化了操作,减少输入错误,提高操作人员的键盘输入效率。然而在 MS-DOS 没有提供具有此功能的命令,这是一个缺憾。为此作者为 MS-DOS 增添一条外部命令 DEFINE,以弥补这一空白。

本文详述 DEFINE.COM 的设计要求、设计思路、具体实现、使用举例和要求。此程序不仅为用户提供了方便,而且还提供一个借助汇编语言对操作系统的资源进行扩充的开发实例。

一、扩充键盘的控制

2.00 版本以上的 MS-DOS 提供了扩充键盘控制功能,允许借助设备驱动程序 ANSI.SYS 对键盘上的任意键字进行重定义。将功能键或其它键字的控制符及所要定义的内容按一定的标准格式组合起来,利用输出定向到屏幕的各种 DOS 输出功能调用,即可完成键字的重新定义。由于 ANSI.SYS 提供的功能,可在由 2.00 以上版本 MS-DOS 支持的 IBM PC/XT 及其所有兼容机上实现,借助此功能开发出的软件有很好的兼容性。

对键盘重新定义所用的控制序列有以下格式:

- (1)ESC [#;#;.....#p
- (2)ESC [#;"string";p
- (3)ESC [#;"string";#;#;.....#p

其中#代表一个 ASCII 码,ESC 是代码为 1BH 的控制符。第一个#是要定义的键字的 ASCII 码。功能键 F1~F10 对应的扩充 ASCII 码为 0;59~0;68。后面的“string”和#分别表示定义的内容和其他键字的 ASCII 码,尾部的小写 p 是控制序列结束标志。

例如,ESC[65;81p 定义键字 A(ASCII 码为 65)为 Q(ASCII 码为 81);ESC[98;115p 定义键字 b(ASCII 码为 98)为 s(ASCII 码为 115);而 ESC[0;68;"dir/p";13p 则定义功能键 F10 为 dir/p 命令后跟一个回车符(ASCII 码 13)。

二、DEFINE 的基本要求及实现

作为 DOS 的外部命令,要求 DEFINE 执行速度快、少占用内存空间;另外,要充分利用操作系统现有的功能和资源。这些只有通过汇编语言程序才能达到。为此,作者选用宏汇编程序设计语言 MASM,借助 DOS 功能调用,进行 DEFINE 的开发。为了减少程序的长度,加快运行时装入内存的速度,汇编程序按 .COM 文件格式要求编写。

DOS 的每条命令作为一个专用的处理程序,其处

理步骤是:1. 获取必需的命令执行参数(含缺省情况);2. 对参数进行分析;3. 当参数符合程序设定的要求时,程序按设计的功能继续执行;否则显示出错信息,然后回到操作系统待命状态。DEFINE 的流程正是按照这些工作步骤设计的。

三、具体实现

根据上述设计思路,结合 ANSI.SYS 的功能,作者编写了汇编源程序 DEFINE.ASM。读者可以使用任何一种编辑程序输入 DEFINE.ASM,然后按顺序执行以下操作:

```
C>MASM DEFINE
C>LINK DEFINE
C>EXE2BIN DEFINE.EXE DEFINE.COM
```

生成的 DEFINE.COM 即可作为一个外部命令在 DOS 下直接使用。

四、命令格式和使用举例

DEFINE 使用的命令格式如下:

```
DEFINE/FUNCNAME/parm1/parm2
```

符号/是命令参数的分隔符,不能缺少。参数 FUNCNAME 是要定义的功能键的名字,采用的形式为字母 F(大、小写均可)后紧随一功能键的序号,如 F1、f10,注意序号的值必须是 1~10 的整数。参数 parm1 是要定义的内容,它可以是任何字符串(包含空格符)或关键字,但不能含斜线符/,此参数的最大长度设定为 69。参数 parm2 是一个标识参数,它表示是否在定义的内容后面接回车符,通常它只有 Y 和 N(大、小写均可)两种选择。

在操作系统下,很多功能键都有其特定的定义。有时为了去除用户的定义,有必要恢复操作系统对这些功能键的原定义,DEFINE 还提供了满足这一要求的功能,只需将参数 parm1 和 parm2 均置为空即可。

下面举例说明命令的使用:

```
(1)DEFINE/F10/DIR A;/Y
```

将功能键 F10 定义成 DIR A;并且后跟一个回车;

```
(2)DEFINE/F1/功能键定义外部命令/N
```

将功能键 F1 定义成字符串“功能键定义外部命令”;

```
(3)DEFINE/F1//
```

此命令将功能键 F1 恢复到操作系统的原定义;

下面还有一些其它的例子:

```
DEFINE/F10/TYPE A;DEFINE.ASM/y
```

```
DEFINE/F7/张红/N
```

```
DEFINE/F3/mov ax,/N
```

```
DEFINE/F5/DBASE JHTJ.PRG/n
```

当未按要求的格式输入参数时,DEFINE 显示出错误信息,并给出一个正确的示例。

运行 DEFINE 后,所设的功能键定义一直保持到

再次使用 DEFINE 重新定义或再次起动 DOS 为止。

五、命令使用要求

在使用 DEFINE 前必须确保：

- (1) 用户的 MS-DOS 版本号为 2.00 或更高；
- (2) 系统盘上必须有设备驱动程序 ANSI.SYS，并且在配置文件 CONFIG.SYS 中应该有安装 ANSI.SYS 的命令语句：DEVICE=ANSI.SYS。

```

LF      EQU 0AH      ;换行的 ASCII 码
CR      EQU 0DH      ;回车的 ASCII 码
SPACE   EQU 20H      ;空格符的 ASCII 码
SLASH   EQU 2FH      ;/符的 ASCII 码
DOLLAR  EQU 24H      ;$ 符的 ASCII 码
;定义宏体
CLS      MACRO        ;清屏
        MOV AL,3
        MOV AH,0
        INT 10H
        ENDM
;
DISP     MACRO NOTE    ;字符串显示
        LEA DX,NOTE
        MOV AH,09H
        INT 21H
        ENDM

```

；程序代码段

```

CODE     SEGMENT PARA PUBLIC 'CODE'
        ASSUME CS: CODE, DS: CODE, SS: CODE, ES: CODE
        ORG 100H
START    PROC NEAR
        JMP MASTER
BELL     DB  07H,07H,"$"
ERROR    DB  CR,LF,'命令格式错误！正确命令格式
        (例):DEFINE/F10/定义功能键/N'," $"
COPYRIGHT DB  '(C)Copyright 1989 年湖北省公路局科研所
        张红',LF,CR," $"
FUNCTION DB  1BH,['0;']
        DB  30 DUP(?)
MASTER:
        CLS          ;清屏
        DISP COPYRIGHT ;显示信息
        CLD          ;设定串操作方向
        MOV SI,80H   ;置命令行参数块起始地址
        XOR AX,AX    ;初始化 AX 寄存器
        LODSB        ;取命令行参数长度存于 AL
        CMP AX,0     ;长度为 0 吗？
        JE ERRDS     ;长度为 0 转错误处理
        MOV CX,AX    ;取命令行参数长度
LABEL1:  LODSB        ;去除命令参数块前的空格

```

```

        CMP AL,SPACE ;
        JNE LABEL2   ;
        LOOP LABEL1  ;
LABEL2:  JCXZ ERRDS   ;命令参数全是空格,出错
        CMP AL,SLASH ;
        JNE ERRDS    ;参数块前导标志错误
        LODSB
        OR AL,20H     ;参数串第一字符化为小写
        CMP AL,'f'
        JNE ERRDS     ;第一字符不是 F 或 f,出错
        LODSB
        CMP AL,'1'    ;判断功能键的序号
        JNE LABEL3    ;判知序号不是 1 也不是 10
        MOV BL,AL
        LODSB
        CMP AL,'0'    ;功能键的序号是 10 吗？
        JE LABEL4     ;判知功能键的序号是 10
        CMP AL,SLASH
        JNE ERRDS     ;功能键的序号>10,出错
        MOV AL,BL
        DEC SI
        JMP LABEL4
LABEL3:  CMP AL,'1'    ;
        JB ERRDS      ;功能键序号<1,出错
        CMP AL,'9'    ;
        JG ERRDS      ;功能键序号>9,出错
        JMP LABEL4
ERRDS:  DISP ERROR    ;显示出错信息
        DISP BELL     ;响铃
        JMP EXIT
LABEL4:  LEA DI,FUNCTION+4
        MOV BL,AL
        CMP AL,'1'    ;
        JE LABEL2     ;功能键 F1
        MOV AL,'6'    ;
        JMP LABEL3    ;其他功能键(F2~F10)
LABEL2:  MOV AL,'5'    ;
LABEL3:  STOSB
        MOV AL,BL
        LEA DI,FUNCTION+5
        CMP AL,'0'    ;
        JE LABEL5     ;功能键 F10
        CMP AL,'1'    ;
        JE LABEL4     ;功能键 F1
        SUB AL,02H    ;其他功能键(F2~F9)
        JMP LABEL6
LABEL4:  MOV AL,'9'    ;
        JMP LABEL6
LABEL5:  MOV AL,'8'    ;
LABEL6:  STOSB

```



```

LODSB ;
CMP AL,SLASH ;
JNE ERRDS ;未找到第二分隔符/,出错
LEA DI,FUNCTION+8
LODSB
CMP AL,SLASH ;后接的是/吗?
JE LABEL8 ;是,转恢复原定义处理
LABEL7: CMP AL,CR ;
JE ERRDS ;后接的是回车,出错
CMP AL,SLASH
JE LABEL9 ;定义已完,转后续处理
STOSB
LODSB
JMP LABEL7
LABEL8 LEA DI,FUNCTION+7
LEA SI,FUNCTION+2
MOV CX,4 ;
LABEL9: LODSB ;
STOSB ;
LOOP LABEL9
JMP LABEL12
LABEL9: PUSH AX ;
MOV AL," " ;
STOSB ;
POP AX
LABEL10: LODSB
OR AL,20H ;将回车标识符转换成小写
CMP AL,"n" ;
JE LABEL12 ;不接回车符
CMP AL,"y"
JNE ERRDS ;后随参数不是Y(y)或N(n),出错
LABEL11: MOV AL,";" ;进行后接回车符处理
STOSB
MOV AL,"1"
STOSB
MOV AL,"3"
STOSB
LABEL12: MOV AL,"p" ;p是控制序列结尾标志
STOSB
MOV AL,DOLLAR
STOSB
DISP FUNCTION ;进行功能键定义
EXIT:
DISP BELL ;响铃提示用户操作完毕
MOV AX,4C00H ;返回到操作系统
INT 21H
START ENDP
CODE ENDS
END START

```

设置 MS-DOS 的功能键

广东汕头 金山中学 何管略

众所周知,在 IBM 型机的 BASICA 系统中,有一组串功能键,如 Alt+L=Locate,Alt+S=Screen 等,非常灵活、方便。我们从实践中体会到,运用 Prompt 命令,也可以在 MS-DOS 3.3 系统下,设置串功能键。下面介绍具体制作程序:

(1)在 CONFIG. SYS 文件中写一条 DEVICE=ANSI. SYS 命令,并保证根目录中含有 ANSI. SYS 文件,使系统支持一系列以 ESC 字符领头的字元组的特殊功能。

(2)键入 Prompt \$n\$g\$e[0;18;"Edlin"p 命令。其中 \$n\$=当前的盘符(也可改用 \$p\$=当前的目录路径),\$g\$=大于号,即">".这个部分放前放后都可以;不过,从整体来说,这个部分却是必不可少的。如果缺少的话,其它引数的功能便无从实现。

\$e\$=ESC 字符。[0;18;=Alt+e 的组合键码。这个码值,大小写状态都相同,仅由扫描位置确定。从左向右,逐个加 1,如 Q=16;从上向下,逐行增加 14,如 A=30。字串"Edlin"就是我们所要定义的符号,大、小写都可以,显示时一律原样照抄;但不接受汉字。字符 p 表示串功能键的定义结束。

注意:以 \$ 领头的特殊字符,如 e,n,g,p 等,都用小写,如果改为大写,其相应的功能自行消失。

(3)Prompt 命令的引数最多可达 127 个字符,所以在同一条命令行的后面,还可继续定义其它的功能键,如 \$e[0;68;"Dir";13p 等。其中 [0;68;是 F10(不是 Alt+F10)键的扩展码,13 是回车键的 ASCII 码。

(4)如果设置的串功能键较多,可再键入一条或几条类似的 Prompt 命令。这些工作,键盘操作难免出错,而且断电关机后功能消失,下次开机又得重做,令人厌烦,因此最宜写成批处理程序。

值得特别注意的是,在同一个批处理程序中,如有两条以上的 Prompt 命令,则只有最后一条有效,在此之前的 Prompt 命令全都无效。因为它们在尚未实现之前,就被后者取而代之了。

怎样解决这个矛盾呢?有多少条 Prompt 命令,就写多少个批处理程序,每个程序只有一条 Prompt 命令。开机之后,依次键入. BAT 程序的主名就行了。

本期特向读者介绍在 DOS 状态下定义功能键的两篇文章。两文的设计风格不同,但都富启发性和教益,相信大家会喜欢。我们还希望来稿介绍设备驱动程序 ANSI. SYS 的详细资料。

编者

编译 BASIC 运行文件与 CCDOS 不相容的解决方法

刘亚兵

IBM-PC 编译 BASIC 由于与解释 BASIC 基本兼容,且运行速度快、占用内存少,深受用户喜爱。然而,它与 CCDOS 却不相容,表现在:其运行文件在运行和返回操作系统后均不能显示汉字。通过分析,笔者找到了解决这一问题的方法。

在带 O 参数编译 BASIC 运行文件 .exe 中,均有如下两段指令:

5899,064E0 B40F	MOV AH,0F
5899,0650 E8B0FF	CALL 0603
5899,0653 BB6900	MOV BX,0069
5899,0656 D7	XLAT
5899,0657 50	PUSH AX
5899,0658 B400	MOV AH,00
5899,065A E8A6FF	CALL 0603
5899,065D 58	POP AX
5899,065E A27200	MOV [0072],AL
5899,0603 55	PUSH BP
5899,0604 56	PUSH SI
5899,0605 57	PUSH DI
5899,0606 CD10	INT 10
5899,0608 5F	POP DI
5899,0609 5E	POP SI
5899,060A 5D	POP BP
5899,060B C3	RET

其中,064e~0650 处指令完成取 CRT 当前显示模式,模式代码存于寄存器 AL 中;0658~065a 处指令完成 CRT 显示模式设置。通过 DEBUG 的运行命令 g 和单步命令 t 跟踪发现:在 0656 处 XLAT 指令将寄存器 AL 置零。这样,再通过 0658~065a 处指令就将 CRT 显示模式设置成了 40×25 黑白字符模式,而 CCDOS 操作系统却要求 CRT 处于图形模式下。因此,只要将 XLAT 指令改成 NOP(空操作)指令,系统就不会使 CRT 进入字符显示模式了。这样,编译 BASIC 的 .exe 文件在运行和返回操作系统后就能显示汉字,从而解决了两者不相容的问题。实践证明:修改是切实可行的,而且对原来程序功能的实现无任何不良影响。

修改步骤(假定文件名为 example.exe):

1. 将 example.exe 换名为 example;
2. 通过 DEBUG 将 example 调入内存,用 S 命令检索出 XLAT 指令的地址,按此地址将其内容由 D7 (XLAT)修改为 90(NOP);

3. 存盘、退出 DEBUG;

4. 将 example 换名为 example.exe。

附修改过程屏幕显示信息(带下划线内容表示应由用户键入)。

A)REN EXAMPLE.EXE EXAMPLE

A)DEBUG EXAMPLE

-r

AX = 0000 BX = 0000 CX = 475C DX = 0000 SP = 00EE

DS = 5854 ES = 5854 SS = 5854 CS = 5854 IP = 0100

5854:0100 4D DEC BP

-S 100 485C D7 50 B4 00

注:检索范围上限为

5854:0A36

CX+100h

-E A36 90

-W

Writing 475C bytes

-Q

A)REN EXAMPLE EXAMPLE.EXE

硬盘的加锁和解锁

大庆石油学院 张文

为保护硬盘中的数据 and 文件的安全,人们想出了很多加密方法。最简单而又安全可靠的是锁住硬盘,使非法者不能进入,而合法人员只要解锁即可使用。

本文给出的程序,是通过修改 DOS 分区表中第 5 字节(操作系统代号),来实现硬盘加锁解锁的。将操作系统代号改为 00H(表示无操作系统),即可锁住硬盘。解锁时,将其改为 01H(12 位 FAT 的硬盘)或 04H(16 位 FAT 的硬盘)即可。

程序中 MOV Buff+1C2H,0H 的 1C2H 表示 DOS 分区表中操作系统代号偏移地址,它也可能是 1D2H、1E2H 或 1F2H(请查看所用硬盘分区表)。解锁程序中 MOV buff+1c2H,04H 的 1c2H,也可能为 1D2H、1E2H 或 1F2H。

1. 加锁:运行加锁程序。当再次启动计算机时,就进入不了硬盘。

2. 解锁:用软盘启动,运行解锁程序,然后热启动,就能使用硬盘了。

1. 加锁程序

code segment

assume cs:code,cs:code,ss:code,es:code

org 100H

begin:JMP start

buff DB 200H dup(?)

start:MOV AX,CS

MOV DS,AX

MOV ES,AX

MOV SS,AX

```

MOV DX,80H
MOV CX,1
MOV BX,offset buff
MOV AX,201H
INT 13H
MOV buff+1C2H,0
MOV DX,80H
MOV CX,1
MOV BX,offset buff
MOV AX,301H
INT 13H
RET
code ends
end begin
2. 解锁程序
code segment
assume cs:code,ds:code,ss:code,es:code
org 100H
begin:JMP Start
buff DB 200H dup(' ')
start:MOV AX,CS
MOV DS,AX
MOV ES,AX
MOV SS,AX
MOV DX,80H
MOV CX,1
MOV BX,offset buff
MOV AX,201H
INT 13H
MOV buff+1C2H,04H
MOV DX,80H
MOV CX,1
MOV BX,offset buff
MOV AX,301H
INT 13H
RET
code ends
end begin

```

键控图形处理程序

徐维科

本程序用 BASIC 语言编写,它采用菜单与键盘控制相结合,可在屏幕任意位置进行彩色绘图,改图,并能将绘制好的图形存盘、或调出重新修改、或打印出来。程序运行的开始及各功能调用结束均返回菜单。操作非常简单方便。

在 GWBASIC 下运行此程序后,先进入菜单提示

区。菜单中有下列功能供选择。选择“1”进入全屏绘图状态。程序定义了键盘上的十三个键为绘图专用键,按其它键均无效。小键盘上周围的数字键为方向键,它可使光点向相应方向运动。Ins 键控制画点或不画点。DEL 键控制是否在屏幕左上角显示光点坐标值。按下 C 键可调换八种颜色。Q 键放弃图形编辑返回菜单状态。E 键是存储键,按下此键后在屏幕左上角提示输入存盘文件名,作出响应后应按回车键。长城 0520C-H 型机的彩色显示缓冲区分红,绿,蓝三个区。所以图象存盘要首先定义文本寄存器为图形区,再分别对三种颜色区的象素以二进制文件的形式分别存入三个文件中。当在菜单提示区中选择“2”则显示“请输入图形文件名?”,输入完文件名后按回车键。本程序采用陷阱功能,防止因输入不存在的文件名而使程序中断运行。图形调入后的修改方法同“1”中所述。当选择“3”或者“4”同样要求输入文件名,若文件存在则显示或打印。当处于显示状态时,按任意键则返回菜单。打印图形时,屏幕硬拷贝结束后自动返回菜单。

用本程序建立的图形文件所占的存储空间比较大,每幅图需 144K 的存储空间,所以最好用硬盘操作。对于只要求单一颜色的图形可采用一个色区对其操作。这样每幅图只要 48K 的存储空间,也提高了存取速度。对程序的参数稍加修改即可用于 IBM-PC/XT 型机及兼容机。本程序在长城 0520C-H 型计算机和紫金 3070 型打印机上通过。

```

10 KEY OFF:CLS:DEFINT A-Z:SCREEN 2:ON ERROR
GOTO 500
20 K$=SPACE$(35)
30 COLOR 7,0:CLS:LINE (0,0)-(640,450),3,BF
40 LINE (164,39)-(466,321),2,BF
50 LOCATE 4,30:PRINT "键盘图形处理程序"
70 LOCATE 7,30:PRINT "1 绘制新图形"
80 LOCATE 9,30:PRINT "2 修改图形"
90 LOCATE 11,30:PRINT "3 显示图形"
100 LOCATE 13,30:PRINT "4 打印图形"
110 LOCATE 15,30:PRINT "5 退出"
130 LOCATE 19,24:PRINT "请选择:"
140 A$=INPUT$(1):LOCATE 19,45:PRINT A$
150 IF A$="1" THEN CLS:GOTO 230
160 IF A$="5" THEN COLOR 7,0:CLS:KEY ON:GO-
TO 520
170 IF VAL(A$)<1 OR VAL(A$)>5 THEN 50
180 LOCATE 21,24:INPUT "请输入图形文件名",B$:
CLS
190 DEF SEG=&HB800:OUT &H3DD,2:OUT &H3DE,
1:BLOAD B$+" ".DTB",0
200 OUT &H3DE,3:BLOAD B$+" ".DTG",0:OUT
&H3DE,5:BLOAD B$+" ".DTR",0
210 IF A$="3" THEN C$=INPUT$(1):GOTO 10
220 IF A$="4" THEN 480
230 X=320:Y=225:Q=0:E=1:C=3:M=7:N=0
240 D$=INKEY$:K=VAL(D$):IF K>0 AND K<4
THEN C=1

```


回溯法及其在编程中的应用

杭州第四中学 高三(3) 陈 磊

有相当多的问题,如迷宫问题,跳马问题以及求一些轨迹问题,并不能运用公式推导或采用循环、递归等方法求得它们的解。解答这类题目只能采用穷举、试探等方法,这就是我们所说的回溯法。

回溯法的基本思想是:为了求得问题的解,先选择某一种可能的情况向前搜索,在搜索的过程中,一旦发现原来的选择是错误的,就返回去加以修改,反复进行这种试探性的选择与返回纠错的过程,直到找到问题的解或者得出问题无解的判断为止。

回溯法是很重要的一种方法,在各类程序设计竞赛中常有它的一席之地,本文试图通过迷宫设计的例子介绍它的简单应用。

既然是个迷宫,当然希望有一定难度,不应该一眼就可以找到通路,最好只有唯一通路。此外,应该充分利用迷宫的面积;如果有一块地方是随便怎样走都走不到的,那等于浪费了一块区域。

古希腊神话传说中曾给出走迷宫的一个办法,即拿着一个线团按下面的规律边走边放线:

- (1)凡是走过的路都铺一条线;
- (2)每到一个岔口先找没走过的路线;
- (3)碰到死胡同,就在返回时铺上第2条线作为标记,以后凡遇到有两条线的路,就不走了。

这个办法可能要走许多弯路,但总能在有限步内走出迷宫。这种方法正是编程时采用的回溯过程。回溯算法是以试探搜索为优先的,即不考虑是否可以求解,先以搜索深度(即找到最终点)为主。

我们结合实例来看看程序的设计。我们把迷宫四周称为边墙,中央的墙称为隔墙,每一个小方格称为小间。好的迷宫应具有如下特征:

1. 每个小间的四个方向中至少有一堵隔墙。
2. 隔墙的每一处都通过其它隔墙与四个边墙之一相连。若几处隔墙不和任一边墙相通,它们必然形成“孤岛”,其结果是在这里出现两条通路。

另外,隔墙的任一处只与四周边墙中的一面相连,如果与两面边墙相连,则将阻断所有从左下至右上角的通路,或是将迷宫面积中的一块游离出去。

根据上面的分析,可以用下面的算法生成合乎条件的迷宫:首先假设迷宫范围内所有的隔墙存在,从迷宫左下角一间出发随机地选一个方向,前进一格,然后要判断两条,一是是否出了边墙,二是前方的这一间是否来过。如果两个回答均为否,就撤除这次穿过的那一堵隔墙,并在进入的小间内留下“到此一游”的标志,如果两个回答有一成立(不可能两者皆成立),就换个方向试试。倘若四个方向都不行,就以退为进,退到左下方一间再走,如果这一间都不行,再退到第二间……

直到每个小间(共 N^2 间)都写上标记为止。

程序变量说明:

A(I) $1 \leq I \leq 2 * N * (N+1)$ 。表示各间与各墙之编号。排列自左到右,自上而下。A(I)=0 表明对应墙存在;A(I)=1,对应墙被撤除。

B(I,J) I,J 分别为小间行、列数。B(I,J)=0,此间未走到;B(I,J)=1 此间已去过。

C(I),D(I) 表示第 I 种前进方向, $0 \leq I \leq 6$ 。I=0,4 表示向上;I=1 或 5 向下;I=2,6 向左;I=3 向右。

U(L),V(L) 当前所在小间坐标。

X,Y 前方欲进入一间坐标。

由于采用回溯法,速度不免受到影响,因此在设计时绘制了表示设置情况的指针。N 为格数,N 越大所需时间越长。

综上所述,利用回溯法必须弄清楚

1. 确定搜索的初始状态,即开始状态。
2. 当搜索到 T 步时,第 T+1 步的各种可能的选择将如何表示。
3. 确定回溯的条件是什么? 并且回溯到前一步之后,如何做另一种选择。
4. 确定有解或无解的条件及如何求出所有解。

```
10 TEXT:HOME:VTAB 12:HTAB 14:PRINT "NUMBER-";:INPUT "":N
```

```
15 IF N < 2 OR N > 22 THEN PRINT CHR$(7):GOTO 10
```

```
20 HGR2:HCOLOR=3:FOR I=0 TO 6:READ C(I),D(I):NEXT
```

```
30 DATA 0,-1,0,1,-1,0,1,0,0,-1,0,1,-1,0
```

```
40 M=N+1:P=M+N:Q=P/2:R=N*N:S=2*M
```

```
* N
```

```
45 GOSUB 300
```

```
50 DIM A(S),B(M,M),U(R),V(R)
```

```
60 B(N,1)=1:U(0)=N:V(0)=1
```

```
70 I=U(L):J=V(L)
```

```
80 E=INT(4*RND(1))
```

```
90 FOR H=E TO E+3:X=I+C(H):Y=J+D(H)
```

```
100 IF X<=0 OR X>=M OR Y<=0 OR Y>=M
```

```
OR B(X,Y)<>0 THEN 140
```

```
110 CO%=0:GOSUB 400:K=K+1:DW%=279/R*
```

```
K:CO%=3:GOSUB 400
```

```
120 U(K)=X:V(K)=Y:B(X,Y)=1
```

```
130 A(P*I+J+Q*C(H)+D(H)/2-Q)=1:L=K:
```

```
T=0:GOTO 70
```

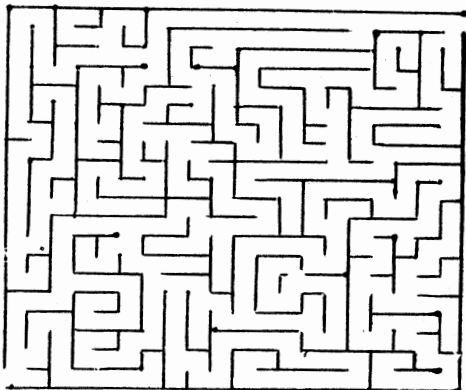
```
140 NEXT
```

```
150 F=F+T:T=1:L=F
```

```

160 IF K<R-1 THEN 70
200 HGR2;HCOLOR=3;A(P)=1;A(S-P+1)=1
210 FOR I=1 TO S
220 IF A(I)<>0 THEN 260
230 Y=INT((I-1)/P)
240 X=I-P * Y;K=X>N;X=X-N * K
250 HPLOT 8 * (X+1-K),8 * (Y+K) TO 8 * X,8
* Y
260 NEXT;GET A$
270 TEXT;HOME;VTAB 12;PRINT "RUN AGAIN? (Y/
N)";
280 VTAB 12;HTAB 34;GET A$;IF A$<>"Y" AND
A$<>"N" THEN PRINT CHR$(7);GOTO 280
290 IF A$="Y" THEN RUN
295 HOME;END
300 HPLOT 1,100 TO 278,100
310 FOR II=4 TO 274 STEP 30
320 HPLOT II,101 TO II,105
330 NEXT II
340 RETURN
400 IF DW% < 5 OR DW% > 274 THEN 450
410 HCOLOR=CO%
420 HPLOT DW%,90 TO DW%,99
430 HPLOT DW% - 3,97 TO DW%,99
440 HPLOT DW% + 3,97 TO DW%,99
450 RETURN

```



中华学习机使用技巧

北京一零一中 任义

一、CATALOG 功能的改进:

使用 CATALOG 命令时,一次可显示 18 个文件名,如果盘中文件超过 18 个,则按任意键可继续列目录。但这样我们有时会忘记以前列过的目标文件名(特别是文件名较长时)。如果对 DOS 系统做些修改,便可使 CATALOG 命令的功能有所增强,即找到目标文件

名后按回车键,系统进入 BASIC 状态,停止列目录,如按其他键则继续列目录。

```

300:20 E3 03 20 D9 03 A9 00 85 48 60
B7EB;00 01 0D
B7F4;01
B7F0;00 20
300G
2039;4C DF BC
B7F4;02
300G
B7EB;00 00 06
B7F4;01
300G
20DF;20 0C FD C9 8D D0 03 4C 2C AE 4C 3C AE
B7F4;02
300G

```

二、巧编音乐程序:

初学编音乐程序的人,往往对音阶、音长代码不熟悉而深感不便。现在向您介绍一种比较方便的方法:先定义两个数组 A(15),B(5),用 A 数组存放音阶代码,用 B 数组存放音长代码,然后 READ 出两个数组的下标,最后 MUSIC A(X),B(Y)就可以了。程序如下:

```

10 DIM A(15),B(5)
20 FOR I=1 TO 5:READ B(I):NEXT;DATA 30,70,
110,160,255
30 FOR I=0 TO 15:READ A(I):NEXT;DATA 1,192,
171,152,140,128,114,102,255,228,205,95,84,75,68,62
40 READ X,Y;IF X<0 THEN END
50 MUSIC A(X),B(Y):GOTO 40
60 DATA 音阶下标(0-15),音长下标(1-5),音阶下标,
音长下标,.....,-1,-1

```

三、DOS 失灵后……

(一)在中华学习机上按复位键后会 DOS 命令失灵,这时我们只须在 BASIC 状态下键入 CALL 1002 或在监控状态下键入 3EAG 即可恢复。

(二)LISA 使用者在 DOS 失灵后,可用 BRK 进入监控,记住 \$1800 的内存值,重新启动 LISA 系统: C600G,再将 \$1800 的值存回,这样程序就不会被清掉。

四、其它

1. 在 LOGO 状态下按复位键进入监控,这时如果键入 CTRL-Y 回车就会回到 LOGO 状态。

2. 有时需要通过 GET 命令进入中文,这样要先有一个空打印,但这会破坏西文格式(屏幕上卷),这时用 PR # 3:CALL 1002 进入中文,既进入了中文系统,又能继续使用 DOS 命令。

3. 从中文系统直接进入低分作图状态,如果您想进入全屏幕作图状态,用 POKE-16298,0;POKE-16302,0;POKE-16300,0;CALL-1998。如果您想进入图形、文本混合状态时用 POKE-16298,0;POKE-16301,0;POKE-16300,0;CALL-1994。

实用改良的 NEW CODE-POKES 程序

黄影昌

当我们在 CEC-I 或 APPLE 机上行进磁盘操作时,有时很想把一个 B 类文件转换成 A 类文件而用 CODE-POKES 程序(很多有关书籍都有介绍)来进行转换。但我在大量实践中发现,该程序最大的缺点是一个本来就不长的文件一经转换就成了一个很长的 A 类文件,因此占用较多的磁盘空间。针对该程序的缺点,我自己试着改良 CODE-POKES 程序,开发了一个新程序;NEW CODE-POKES,经过实验,效果挺不错。下面是该程序的几点说明:

20 句,输入转换后文件的文件名。

30 句,输入 B 类文件的首地址。

40 句,输入 B 类文件的末地址。

50 句,输入转换后 A 类文件的起始行号。

55—190 句,是本程序的转换部分。

具体的操作过程如下:

1. 调入需转换的 B 类文件,查出其首,末地址。

首地址: PRINT PEEK (43635) * 256 + PEEK (43634)✓

末地址: PRINT PEEK (43635) * 256 + PEEK (43634) + PEEK (43617) * 256 + PEEK (43616) - 1✓

2. 运行 NEW CODE-POKES,光标出现后,再 EXEC 文件名✓

待光标再现后,内存中就是转换后的 BASIC 程序了,此时你可将它存入磁盘。

```
10 D$ = CHR$(4);C=1
20 TEXT;HOME;VTAB 4;INPUT "FILENAME:";N$
30 INPUT "SHOU DI ZHI:";A
40 INPUT "MO DI ZHI:";B
50 INPUT "QI SHI HANG HAO:";LN
55 PRINT D$ "OPEN";N$
60 PRINT D$ "DELETE";N$
70 PRINT D$ "OPEN";N$
75 PRINT D$ "WRITE";N$
80 FOR P=A TO B
100 IF C=61 THEN P=P+1;C=1
110 IF C(>) 1 THEN 150
120 PRINT;PRINT LN;"DATA";
130 LN=LN+1
150 PRINT PEEK(P);;C=C+1;IF C=61 THEN 100
160 IF P<B THEN PRINT";";
170 NEXT P
175 PRINT
180 PRINT LN+1;"FORI=";A;"TO";B;"READC;
```

POKEI,C;NEXTI"

```
190 PRINT D$ "CLOSE";N$
200 DEL 10,200
```

巧解“八皇后”问题

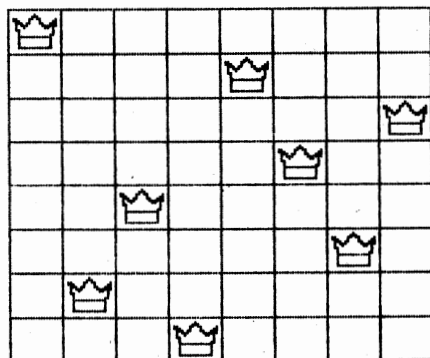
南京师范大学附中高一(1) 柳杨

“八皇后”问题是伟大的数字家高斯提出来的。它要求在国际象棋盘中摆下的八个皇后互不相吃。

本程序求解“八皇后”问题,并找出所有 92 种解答。程序运行过程中还配有生动的移子画面。

附程序清单和一种解答的拷贝。

```
5 DIM C(8),T(8),S(15),P(15);D=1
10 HOME;HGR2;HCOLOR=3
20 FOR I=0 TO 160 STEP 20,HPLLOT I+60,0 TO I+60,160,HPLLOT 60,I TO 220,I;NEXT
30 K=1
40 IF T(K)=1 THEN 80
50 IF S(8+D-K)=1 THEN 80
60 IF P(K+D-1)=1 THEN 80
70 C(D)=K;T(K)=1;S(8+D-K)=1;P(K+D-1)=1;CO=3;GOSUB 600;D=D+1;IF D>8 THEN TA=TA+1;GET A$;GOTO 90
75 GOTO 30
80 K=K+1;IF K<=8 THEN 40
90 D=D-1;IF D<1 THEN TEXT;PRINT "TOTAL=";TA;PRINT" * * * END * * *";END
100 K=C(D);CO=0;GOSUB 600;T(K)=0;S(8+D-K)=0;P(K+D-1)=0;GOTO 80
600 X=64+(K-1)*20;Y=12+(D-1)*20
610 HCOLOR=CO
620 HPLLOT X,Y TO X+12,Y TO X+12,Y+5 TO X,Y+5 TO X,Y TO X-2,Y-8 TO X+2,Y-4 TO X+6,Y-10 TO X+10,Y-4 TO X+14,Y-8 TO X+12,Y
640 RETURN
```



关于 APPLE 和 CEC-I 机数值计算误差的讨论

对《APPLE 和 CEC-I 设计者的疏忽》一文看法

成都人防电器设备厂 刘峰

贵刊 90 年 8 期发表了刘光明的《APPLE 和 CEC-I 设计者的疏忽》一文,作者在打印数字“8.13”时产生了误差。这种误差常常是难以避免的,它并非是设计者的疏忽,而是由于计算机内采用二进制数,而高级语言均使用十进制数,所以程序和数据输入计算机就有一个数制转换的问题。一般的整数只要不超过机器允许的长度,在数制转换时不会发生误差,但是小数就不一样了,小数的数制转换多数只能取近似值,比如十进制小数 0.1 化为二进制小数是 0.00011001100……。这是一个无限循环小数,必须经取舍后成为近似值。其次,计算机用固定的字节数存放每一个实型数,而在输出时又仅仅以前几位有效数字的形式打印出来,这常常使得二者不可能完全一致。APPLESOFT 以 5 个字节存放一个实型数,输出最多 9 位有效数字,因此在两数相减时,只要差的有效数字不足 9 位,内存中的隐含误差部分便向前递补,凑足 9 位数字输出。当然,并非所有实型数在数制转换时都会发生误差。当其小数部分为 0.5 的正整数次幂时,数制转换时不会发生误差。另外,误差极小时,机器也会忽略不计。

其实,乘方与开方等运算均可导致误差,特殊情况下误差还相当大。当相差很大的两个整数相加减时,大数会“吃掉”小数。例如计算机计算“ $2.54E9+5$ ”时,答案还是 $2.54E9$ 。运算乘方与开方时所产生的误差是由于计算机的系统软件采用与人工计算完全不同的近似算法(迭代逼近法)所引起的,而相差很大的两个整数相加减时则是因为两个数的阶码值相差较大引起的,如把第一个数换为定点数,情况就不会一样了。

很明显,无论是实型小数的数制转换引起的误差,还是整实数的近似计算误差,都会给运算带来不良影响,甚至使程序无法继续运行。那么,进行实型小数的计算和乘方开方的计算,都要设法消除误差。一般的方法是测出原始数据的小数位中最长的一个(用对数和字符串处理),在输出前利用 INT(X) 函数实现必要的舍入。其次是在用小数作累加时,可利用整数取代小数作为循环变量值并增加运算,这样误差值可显著降低。对于避免等值比较时逻辑值为零的情况判断失误,可采用限定误差范围的方法,如用 $IF\ INT(SQR(X))-SQR(X)<1E-6\ THEN\ PRINT\ X$ 可顺利地判断 X 是否为完全平方数。例如打印“8.13”时,可采用 PRINT

(INT(8.13 * 100)/100)或 PRINT “8.13”等。

APPLE 缺陷的解决办法

福建邵武市闽北地质队 陈盛旋

产生类似问题的数还有 8.104、-8.13 等等,这个问题虽无大错,却破坏了正确的打印格式、给编程者造成不大不小的麻烦。一个简易的解决办法,是在打印输出之前对数据进行四舍五入和小数点截位处理。具体公式如下:

$$A = -INT(-1 * A * 1E2 + 0.5) / 1E2$$

在打印变量 A 之前,先进行上述公式计算,即可将 A 的小数点经四舍五入后只保留两位,即使 8.13000001 也将打印成 8.13 出来。这就解决了刘光明同志亟待解决的问题。

这个公式对正数负数和零均有效,因而也可用于普遍的数据四舍五入处理。如果要保留三位小数或四位小数,只需将公式中的 1E2 改成 1E3 或 1E4 即可。这一公式解决了 APPLE 机没有四舍五入语句的缺陷。

附:示范程序:

```
10 A=8.13
20 PRINT "A=";A
30 A=-INT(-1*A*1E2+0.5)/1E2
40 PRINT "A=";A
50 END
```

JRUN

A=8.13000001(原来的 A 值)

A=8.13(经过处理的 A 值)

数值计算中存在问题的解决办法

吉林市水泵厂 朱洪波

笔者在实践中,找出了解决问题的简单办法。只要在程序中增加一条语句:

```
10 A=8.13
20 A=INT(A*100)/100
30 PRINT A
RUN
8.13
在下面的计算中也会出现类似现象
10 M=60.07;N=4
20 PRINT M+N
```

RUN

60.0700001

解决办法也跟前面一样,增加下面一条语句:

P=INT((M+N)*100)/100

经过上述处理后,就不会出现错误的结果了。

笔者在实际工作中,发现 APPLE 及其兼容机还存
在下面问题,即在小数部分取舍时,出现错误的计算结果。例:

10 P=34.555

20 PRINT(P*100+0.5)/100

RUN

34.55

打印的结果应该是 34.56,而经过上述计算却是
34.55。这种现象的出现,给软件的编制带来一定的麻
烦,特别是在财务管理中,更是不允许出现。笔者曾看
到用下列方法来解决,例如:

10 P=0.525

10 PRINT INT(P*100+0.5000001)/100

RUN

0.53

但此方法没有从根本上解决小数的取舍问题。

例:

10 P=34.555

20 PRINT INT(P*100+0.5000001)/100

RUN

34.55

显然结果还是不对。笔者在编制程序中使用下面
的改进方法,取得了较好的效果,该程序消除了一切在
保留小数中出现的问题,从根本上解决了小数取舍时
产生的错误。程序清单如下:

10 INPUT "A,N=";A,N

20 GOSUB100;PRINTA;GOTO10

100 A\$=STR\$(A);L=LEN(A\$)

110 FORI=1 TO L

120 IF MID\$(A\$,I,1)="." THEN

I1=I;GOTO 130

125 GOTO 140

130 I=I1+N+1;A1=VAL(MID\$(A\$,I,1))+
5/10^(N+1);A=VAL(MID\$(STR\$(A1),1,I-
1));RETURN

140 NEXT I;RETURN

在上面的程序中,A 为要取舍的小数,N 为保留小
数的位数,120 语句是测量在第几位出现小数点,130
语句是根据测得的位数计算出要取舍的结果。例:

RUN

A,N=? 34.555,2

34.56

A,N=? 0.525,2

0.53

A,N=? 0.5265,3

0.527

该程序虽然较长,但是只要将 100~140 语句作为
一个子程序使用就比较方便了,例:

10 A=X;GOSUB100;X=A

可以看到,上面的例子是非常简单的,对于一个或
多个数组变量也能方便地解决如:

10 FOR I=1 TO 30:A=A(I);GOSUB100;

A(I)=A;A=B(I);GOSUB100;B(I)=A;

NEXT

前面提出的问题,是 APPLE 及其兼容机上固有
的。对于现有的机器,要想使之运行出准确的结果,只
有从编程上采取措施。这样,苹果机还是能在各行各业
中发挥作用的。

无数据磁盘格式化程序

呼和浩特市第一中学九二(3)班 姜宏岩

当我们使用 DOS 的 INIT 命令对磁盘初始化时,
命令的处理过程分为以下四个步骤:

(1)格式化磁盘。将磁盘划分为 \$0~\$22 共 35
条磁道,每条磁道有 \$0~\$F 共 16 个扇区的格式。

(2)把 \$11 磁道的 \$0 扇区定义为盘卷内容表,或
称 VTOC 表。VTOC 表上记载了磁盘中每个扇区是
“空”还是“被使用”的状态。将 \$11 磁道的其余扇区
\$F~\$1 定义为供磁盘文件目录使用,然后把磁道
\$0~\$2 及 \$11 在 VTOC 表中置为“被使用”。

(3)从内存中把 DOS 代码写入 \$0~\$2 磁道。

(4)把内存中的 BASIC 程序存入磁盘,程序名要
在 INIT 命令中写明(一般起名 HELLO)。

因此,在已初始化的磁盘中,DOS 占了 3 个磁道,
VTOC 表和文件目录共占 1 个磁道,再扣除 HELLO 程
序所占的空间后,实际剩下的空间就更少了。事实上,
有时并不需要每个磁盘都存放 DOS 代码,尤其是一些
用于存放数据的磁盘。下面所提供的程序,运行后将修
改内存中 DOS 的 INIT 命令程序,只对磁盘进行格式
化,不写入 DOS 及 HELLO 程序,而将 \$1、\$2 磁道释
放出来给用户使用,\$0 磁道由于有其它原因仍然不
能使用。

100 TEXT;HOME;VTAB 3

110 INVERSE;HTAB 8

120 PRINT "DATA DISK FORMATTER"

130 NORMAL;PRINT

140 POKE 42347,96;POKE 44723,4

160 POKE 44799,76;POKE 44800,127;POKE 44801;

179

170 PRINT;PRINT

180 PRINT "INSERT DISKETTE TO BE INITIALLIZED"

190 PRINT "PRESS [RETURN] TO PROCEED"

200 PRINT "[ESC] TO EXIT"

210 PRINT "BEGIN(Y/N)?"

220 GET A\$

230 IF A\$ = "Y" THEN 250

240 IF A\$ = "N" THEN 260

250 PRINT CHR\$(4); "INIT XXX,254"

260 POKE 42347,76;POKE 44723,12;POKE 44799,32;
POKE 44800,194;POKE 44801,183;END

跟踪机器语言 程序的运行

天津 武清 杨村一中 田俊成

苹果机在监控状态下没有跟踪运行命令,这给调试机器语言程序带来一定程度的不便。为此我们用 6502 机器语言编了一段程序,在监控状态下跟踪机器语言程序的运行,可以观察机器语言的执行过程,特别对程序出现死循环等问题,可以很快找到。

使用时,首先在监控下键入 3F9:00 10,然后每次调用只要键入:[被执行程序首地址] CTRL-Y 则从指定地址开始每执行完一条指令后,显示该指令。在执行过程中用 CTRL-S 使之暂停,再次使用 CTRL-S 则继续执行,如此可停下来查看当前提示。

```
1000- 20 75 FE A9 B1 85 48 A9
1008- EA 8D CA 10 8D CB 10 20
1010- D0 F8 A0 00 B1 3A C9 10
1018- F0 2F C9 30 F0 2B C9 50
1020- F0 27 C9 70 F0 23 C9 90
1028- F0 1F C9 B0 F0 1B C9 D0
1030- F0 17 C9 F0 F0 13 C9 4C
1038- F0 1A C9 6C F0 27 C9 20
1040- F0 3B C9 60 F0 57 4C BB
1048- 10 8D C9 10 A9 15 8D CA
1050- 10 4C C6 10 C8 B1 3A 85
1058- 3C C8 B1 3A 85 3B A5 3C
1060- 85 3A 4C 07 10 C8 B1 3A
1068- 85 3C C8 B1 3A 85 3D A0
1070- 00 B1 3C 85 3A C8 B1 3C
1078- 85 3B 4C 07 10 18 A5 3A
1080- 69 02 85 3C A5 3B 69 00
1088- 48 A5 3C 48 C8 B1 3A 85
1090- 3C C8 B1 3A 85 3B A5 3C
1098- 85 3A 4C 07 10 68 85 3A
10A0- 68 85 3B E6 3A D0 02 E6
10A8- 3B A5 3A C9 85 D0 09 A5
10B0- 3B C9 FF D0 03 4C 85 FF
10B8- 4C 07 10 88 C8 B1 3A 99
10C0- C9 10 C4 2F D0 F6 20 3F
10C8- FF AE EA EA 08 85 45 86
10D0- 46 84 47 68 85 48 20 53
10D8- F9 85 3A 84 3B 4C 07 10
10E0- E6 3A D0 02 E6 3B A0 00
```

10E8- B1 3A 85 2F 20 53 F9 85

10F0- 3A 84 3B 4C 07 10

局部清屏程序

解放军后勤工程学院 程华

用户在编制绘图程序时,时常要清除部分图形而保留另一部分图形。遇到这类情况一般是用 BASIC 作图语句逐点清除,运行速度受到限制,不利于实时应用。本文提供一段 6502 机器语言子程序,能方便的实现局部清屏,清屏面积的大小可由用户在指定单元内输入数据来实现。它可完成小到一个光点,大到整个屏幕的清除。此程序在 APPLE II 及其兼容机中均可使用。现将使用方法简介如下:

由于本程序中各种转移指令均采用相对寻址,所以程序可根据需要置于内存任何位置。

应用此程序时如果在第一图形页清屏,需将 \$20 (等于十进制 32,以下用 L 表示十进制),第二页为 \$40(L64)置于 \$E6 单元(L230)。然后将需清屏的起始水平坐标值高位置于 \$FA(L250),低位置于 \$FB(L251)。所谓高位是指水平坐标值除以 256 时的整数部分,余数即为低位值。将终点水平坐标值高位置于 \$FC(L252),低位置于 \$FD(L253),将起点的垂直坐标值置于 \$FE(L254),终点的垂直坐标值置于 \$FF(L255)。完成后,在 BASIC 程序中,可用 CALL768 调用此程序;在汇编语言中可用 JSR \$300 调用。

请注意:坐标原点设在屏幕的左上角(0,0)。应用时起点坐标值必须小于或等于终点坐标值。另外终点坐标值不要大于图形页的尺寸,即水平不大于 279,垂直不大于 191。下面举例子说明程序的用法:

如果已定义好图形页并已调入一图形,现想清除中间一部分,其水平坐标起点是 100,终点是 170,垂直坐标起点是 50,终点是 85。可编制下列 BASIC 程序:

```
1000 POKE 250,0;POKE 251,100;POKE 252,0;
POKE 253,170;POKE 254,50;POKE 255,85;CALL
768
```

运行 1000 句后可看到,所要求清除部分的图形已被清除,而其余部分完好无损。

```
0300- A5 E4 48 A2 00 86 E4 E6
0308- FF A5 FE 85 F2 A2 00 B5
0310- FA 95 06 E8 E0 04 90 F7
0318- A6 07 A4 06 A5 F2 20 57
0320- F4 A5 06 C5 08 90 14 A5
0328- 07 C5 09 90 0E E6 F2 A5
0330- F2 C5 FF 90 D8 C6 FE 68
0338- 85 E4 60 E6 07 D0 D9 E6
0340- 06 18 90 D4
```


扩展 BLOAD 命令功能

重庆大学 蒋阳

在 DOSS3.3 磁盘操作系统中,有一个 BLOAD 命令,其功能是将存于磁盘上的二进制(B类)文件的内容调入内存。对一个不知其功能或加密的 B 类文件,我们为了读懂或解密,首先必须知道它存放于内存的首地址、长度或者末地址,而 BLOAD 命令不能直接提供给用户。为此本人对 BLOAD 命令作了改进,当用户使用改进后的 BLOAD 命令调 B 类文件时,既完成原 BLOAD 的功能,同时自动清屏,并在屏幕的第一行上从左至右显示出文件放于内存的首地址、长度、末地址。

```
5 HOME
10 FOR I=48351 TO 48369:READ A;POKE I,A;NEXT
13 FOR I=46771 TO 46841:READ B;POKE I,B;NEXT
15 FOR I=47721 TO 47762:READ C;POKE I,C;NEXT
20 PROKE 41861,76;POKE 41862,179;POKE 41863,
```

182

```
30 DATA 165,7,233,0,32,218,253,165,6,32,218,
253,40,174,114,170,76,136,163
40 DATA 8,32,88,252,169,211,32,240,253,169,189,
32,240,253,169,164,32,240,253,173,115,170,32,218,
253,173,114,170,32,218,253,169,160,32,240,253,169,
204,32,240,253,169,189
45 DATA 32,240,253,169,164,32,240,253,173,97,
170,32,218,253,173,96,170,32,218,253,169,160,32,
240,253,76,105,186
60 DATA 169,197,32,240,253,169,189,32,240,253,
169,164,32,240,253,24,173,114,170,109,96,170,133,6,
173,115,170,109,97,170,133,7,56,165,6,233,1,133,6,
76,223,188
```

从“重赏”谈起

山东金乡五中 卢保堂
中国人民公安大学 卢玉堂

印度有一个饶有趣味的故事。传说舍罕王打算重赏国际象棋的发明人——宰相西萨·班·达伊尔,国王问他有何要求,这位宰相说,只需在棋盘第一格放一粒麦子,第二格放两粒,第三格放四粒,这样依次加倍地放下去,直到放完第六十四格,把这些麦子都赏给他就可以了。国王认为这是区区赏金,微不足道,就满口答应,结果当然是绝对办不到的。如果我们估算一下,

就会发现这个数字很大,可以用电脑帮助解决这个问题。程序一运行结果表明,此数是全世界两千年内生产的全部麦子。由于受计算机位数的限制,小麦粒数的后几位无法显示。程序二能精确地显示出小麦粒数。

程序和变量简单说明如下:

第 5 句:输入棋盘格数

第 6 句:计算总麦粒数的位数

第 10~80 句:清变量

第 100 句:赋初值

第 100~300 句:计算第 I 格内与前 I 格内总和的麦粒数

第 500~700 句:打印输出

S(I,J)表示第 I 格内麦粒数第 J 位的数字。

H(J)表示总麦粒数第 J 位的数字

SJ,JS 分别表示第 I 格内第 J 位的进位标志和进位

HJ,JH 分别表示总麦粒数第 J 位的进位标志和进位。

程序一

```
5 INPUT "QI PAN GE SHU?";N
10 S=1;A=1;N=N-1
15 IF N=0 THEN 50
20 FOR I=1 TO N
30 A=2*A;S=S+A
40 NEXT I
50 PRINT "S=";S
60 END
]RUN
QI PAN GE SHU ? 64
S=1.84467441E+19
]RUN
QI PAN GE SHU ? 73
S=9.44473297E+21
```

程序二

```
5 INPUT "QI PAN GE SHU?";N
6 N1=INT (LOG (2^N)/LOG (10))
7 N=N-1
10 DIM S(N,N1),H(N1)
20 FOR I=0 TO N
30 FOR J=0 TO N1
40 S(I,J)=0
50 NEXT J;NEXT I
60 FOR J=0 TO N1
70 H(J)=0
80 NEXT J
100 S(0,0)=1;A=0;H(0)=1
105 IF N=0 THEN 500
110 FOR I=1 TO N
120 K=I-1
130 FOR J=0 TO N1
140 IF S(K,J)<5 THEN SJ=0;S0=S(K,J);GOTO 160
150 SJ=1;S0=S(K,J)-5
160 S(I,J)=S0*2+JS
170 IF SJ=0 THEN JS=0;GOTO 190
```

```

180 JS=1
190 NEXT J
200 FOR J=0 TO N1
210 H0=S(I,J)+H(J)
220 IF H0<10 THEN LET HJ=0;GOTO 250
230 HJ=1;H0=H0-10
250 H(J)=H0+JH
260 IF HJ=0 THEN LET JH=0;GOTO 280
270 JH=1
280 NEXT J
290 HJ=0;SJ=0
300 NEXT I
500 PRINT
550 PRINT "S=";
600 FOR J=N1 TO 0 STEP -1
610 PRINT H(J);
620 NEXT J
630 PRINT
700 END

```

```

]RUN
QI PAN GE SHU ? 64
S=18446744073709551615
]RUN
QI PAN GE SHU ? 73
S=9444732965739290427391

```

在 APPLESOFT 下使用小汇编

南京市五十中 嵇克威

大家都知道,在苹果机的整数 BASIC 下可以使用小汇编。只要机器从整数 BASIC 状态进入监控,再键入 F666G 即进入小汇编。小汇编系统短小精悍,简单实用,深受广大用户喜爱。可是目前苹果机多在 APPLESOFT BASIC 下工作,无法使用小汇编。为了解决这个问题,我将小汇编程序代码搬移到较低地址的 RAM 空间,对程序内部绝对寻址指令的高位地址作相应修改,具体步骤如下:

(上接 26 页)

(3)C 语言除了有优先级外,还有结合性问题。所谓结合性是指当表达式中同时出现多个优先级相同的运算符时的求解顺序。如果运算顺序是从左到右,叫左结合性;从右到左,叫右结合性。表中最右边一项,说明了各级运算符的结合性。

(4)C 语言中有 5 个运算符的结合性是可交换的。它们是 *、+、&、|、^。

(5)此外,注意 Turbo. c 表达式中允许随意增加空格和使用多余的圆括号。

①引导 DOS3.3 主盘,载入 DOS 和 INTBASIC
②键入 INT 进入整数 BASIC;CALL-151 进入监控。

③在监控下键入:7500(F500.F63CM

④键入以下命令:

* 7537:75 N 755B;75 N 75BF;76 N 75DD;76N 75E7:76 N 7633;75

⑤键入 CTRL-C 回到整数 BASIC,然后输入 FP 转到 APPLESOFT 状态。

现在可以在 APPLESOFT 下使用小汇编了。要使用时,只要进入监控,再输入 7592G。为了更方便使用,可把 & 向量指向改造后的小汇编,即键入:3F5:4C 92 75

以后只要键入 "&"即可进入小汇编。

⑥为了保存改造后的小汇编,先键入 74FD:4C 92 75 然后存盘

BSAVE MINIASM,A\$74FD,L\$140

以后只要 BRUN MINIASM 就能进入小汇编。

工资票面张数计算 程序的简化

浙江奉化一中 孙承灿

本刊 90 年第四期的《一个简单实用的工资票面张数计算程序》一文的程序,本人认为过于繁杂。其实,当一个单位 N 个人的应发工资数存放在数组 N(I)中时,只要在程序的开头加上第 5 行,在适当的位置插入一句 GOSUB 1000 即可。程序如下:

```

5 DIM Y(12),X(12);FOR I=1 TO 12;READ X(I);
NEXT;DATA 10000,5000,1000,500,200,100,50,20,10,5,
2,1
100 GOSUB 1000
:
:
1000 FOR I=1 TO N;A=N(I)*100;FOR J=1 TO 12;Y
= INT(A/X(J));Y(J)=Y(J)+Y;A=A-Y*X(J);NEXT;
NEXT;RETURN

```

[练习 4.15]

main ()

```

{ printf("%d\n",5>&&!(1<5)||2<=3);
printf("%d,%d\n",5+3,3+5);
printf("%d,%d\n",5|3,3|5);
}

```

A)EXP4-15

1

8,8

7,7

第四讲 C 运算符

李文兵

运算符较多,是 C 语言的一个显著特点。那么,C 运算符究竟有多少,又如何使用,请看本讲。

1 算术、逻辑运算符

(1)算术运算符 标准算术运算符见表 4.1。

表 4.1 算术运算符

运算符	操 作	项 数	优先级
-	负	单项	1
*	乘	双项	2
/	除	双项	2
%	余	双项	2
+	加	双项	3
-	减	双项	3

[练习 4.1]

```
A>type exp4-1.C
main()
{ int i=2,j=3;
  printf("add=%d\n",i+j);
  printf("mod=%d\n",(i+j)%3);
}
A>exp4-1
add=5
mod=2
```

该程序在 (i+j)%3 中使用了运算符(),它是优先级最高的运算符,常用它改变表达式的运算顺序。

(2)关系运算符 是用来判断两个值大小的运算符,共有六种,如表 4.2 所示。

表 4.2 关系运算符

运算符	操 作	项 数	优先级
>	大于	双项	1
>=	大于或等于	双项	1
<	小于	双项	1
<=	小于或等于	双项	1
==	等于	双项	2
!=	不等于	双项	2

这些运算符的用法是:

操作数 1 关系运算符 操作数 2

该式成立时,运算结果为 1(代表真);否则为 0(代表假)。注意,这里的 1 或 0 不是布尔量,是整型常数 1 或 0。C 语言里没有定义布尔量。

[练习 4.2]

C>TYPE EXP4-2C

```
main()
{ int i=5,j=3,a=3,b=5;
  printf("< <= > >= == !=\n");
  printf(" %d %d %d %d %d %d\n",
    i<j,i<=j,i>j,i>=j,i==j,i!=j);
  printf(" %d %d %d %d %d %d\n",
    5<3,5<=3,5>3,5>=3,5==3,5!=3);
  printf(" %d %d %d %d %d %d\n",
    a<b,a<=b,a>b,a>=b,a==b,a!=b);
}
```

C>EXP4-2

<	<=	>	>=	==	!=
0	0	1	1	0	1
0	0	1	1	0	1
1	1	0	0	0	1

(3)逻辑运算符 基本的逻辑运算符有逻辑与(AND)、逻辑或(OR)和逻辑非(NOT)。在 C 语言中其运算符号,如表 4.3 所示。

表 4.3 逻辑运算符

运算符	操 作	项 数	优先级
&&	逻辑与	双项	2
	逻辑或	双项	3
!	逻辑非	单项	1

C 语言无布尔量,故逻辑运算结果也用整数 1 代表真,整数 0 代表假,而对于操作数来说,若其值不是 0,为真;否则为假。

[练习 4.3]

A>type exp4-3.c

```
main()
{ printf("%d\n",2&&1);
  printf("%d\n",2||3);
  printf("%d\n",!2);
}
```

A>exp4-3

1
1
0

(4)位逻辑运算符 有六种,如表 4.4 所示

表 4.4 位逻辑运算符

运算符	名 称	操 作	项 数	优先级
&	位积	按位与	双项	3
	位和	按位或	双项	5
^	位差	按位异或	双项	4
-	位非	按位取反	双项	1
<<	左位移	按右值位数左移左值	双项	2
>>	右位移	按左值位数右移左值	双项	2

〔练习 4.4〕

C>type exp4_4.c

main()

```

{ int x=7,y=5;
  printf("&| ^ - << >>\n");
  printf("%x %x %x %x %x %x",
        x&y,x|y,x^y,-x,x<<y,x>>y);
}

```

c>exp4_4

```

&| ^ - << >>
5 7 2 fff8 e0 0

```

2. 递增、递减、赋值运算符

这三种运算符用在表达式中,使其具有副作用。

(1)递增与递减运算符 这两种运算符是 C 语言特有的,它们都是单项运算符,用法如下:

x++ 等价于 x=x+1

x-- 等价于 x=x-1

这两种运算符可放在操作数的前面,叫前置运算,也可放在操作数之后,叫后置运算。

对于前置运算,例如:

x=10;y=++x;

运算结果为:

x=11;y=11。

对于后置运算,例如:

x=10;y=x++;

运算结果为:

x=11;y=10。

〔练习 4.5〕

C>TYPE EXP4-5.C

main()

```

{ int i=10,j;
  j=i;
  printf(" i\t j\n");
  printf("----\t----\n");
  printf(" %d\t %d\n",i++,++j);
  j=i++;
}

```

printf("%d\t %d\n",i++,j++);

}

C>EXP4-5

```

i      j
----  ----
10     11
12     11

```

(2)赋值运算符 我们已经知道,C 语言的赋值运算符是 "=",其用法是:

变量=表达式;

此外,在 C 语言中,假定 a 为变量,e 为表达式,op 为运算符,则表达式:

a=a op e;

可用如下表达式替代。

a op=e;

其中 op= 叫做复合赋值运算符。op 可以是 +、-、*、/、%、&、|、^、<<、>>,共 10 个。这样,赋值运算符又扩展了 10 个。

〔练习 4.6〕

C>TYPE EXP4-6.C

main()

```

{ int a=100,b=15;
  a+=b;
  a-=b;
  a*=b;
  a/=b;
  a%=b;
  a<<=4;
  printf("shift left %d\n",a);
  a>>=4;
  a&=b;
  a|=b;
  a^=0xffff;
  printf("exclusive or %d\n",a);
}

```

C>EXP4-6

shift left 160

exclusive or -16

3 求字节数、条件运算、逗号

(1)sizeof 运算符 其功能是计算变量或类型的大小,其用法如练习 4.7 所示。

〔练习 4.7〕

C>TYPE EXP4-7.C

main()

```

{ int a;
  int b[100];
  printf("1: sizeof int variable=%d\n",
        sizeof(a));
  printf("2: sizeof int=%d\n",
        sizeof(int));
  printf("3: sizeof array of int=%d\n",

```



```

        sizeof(b));
printf("4: sizeof float=%d\n",
        sizeof(float));
printf("5: sizeof double=%d\n",
        sizeof(double));
}

```

C>EXP4-7

```

1: sizeof int variable=2
2: sizeof int=2
3: sizeof array of int=200
4: sizeof float=4
5: sizeof double=8

```

(2)条件运算符 C语言特有,也是C语言中唯一的三项运算符,功能相当于一段程序,用法如下:

表达式1? 表达式2:表达式3

其含义是:

若表达式1为真,则整个表达式的解就是表达式2的解,否则,为表达式3的解。

[练习4.8]

C>TYPE EXP4-8.C

```

main()
{
    int a=5,b=3;
    printf("(a>b? a:b=%d\n",
            (a>b)? a:b);
}

```

C)EXP4-8

(a>b)? a:b=5

(3)逗号运算符 该运算符用在括号中有多个表达式的场合。其功能是从左到右求解各个表达式,而整个表达式的值为最后所求解的表达式。例如:下面逗号表达式的解为x的值等于4。

x=3,x=4;

逗号运算符有效地用于while语句和for语句中。

[练习4.9]

C>type exp4-9.c

```

main()
{
    int x,y;
    printf("%d\n", (x=3,y=4));
}

```

C)EXP4-9

4

4 指针的概念及其运算符

(1)指针的概念 指针也是一种变量,是用来指定另一个变量(允许是结构变量,另一个指针变量)所在场所的变量,其内容为所指定变量的地址,指针变量的定义格式如下所示:

数据类型 * 指针变量名;

例如:

char * p; p为指向字符型数据的指针。

int * var var为指向整型数据的指针。指针变量p和var的内容及其功能如图4.1所示。

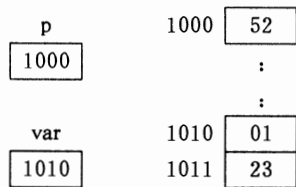


图4.1 指针的概念

(2)指针有关的运算符 与指针有关的运算符有&与*两种。&用来求变量的地址,叫地址运算符。*用来访问指针所指变量的内容,例如:

int i=10; 定义变量i且赋初值10

int * p; 定义p为指针变量

p=&i; 给p赋上变量i的地址,这时,就可以说指针p指向了变量i。

*p=21; 给p所指的变量内赋上数据21。

说明上述分析正确性的程序,见练习4.10

[练习4.10]

c>TYPE EXP4-10.C

```

main()
{
    int i=10,j;
    int * p;
    p=&i; j= * p; * p=21;
    printf("address of p j i * &i \n");
    printf("%p %d %d %d \n",
            p,j,i, * &i);
}

```

C)EXP4-10

```

address of p j i * &i
FFDA 10 21 21

```

5 类型显示转换运算符

C语言中的数据类型转换分为隐式(自动进行的)和显式(强制性的)两种。

(1)隐式转换 C编译程序对表达式中出现的不同数据类型,按如下规则自动进行类型转换。

①表达式中的所有float类型数据,在任何操作之前均被转换为double类型。只有在数组和结构中float类型才真正按float型处理

②在算术表达式里,所有short和char型数据都转换为int型。

③参加运算的两个操作数类型不同时,要按图4.2所示把优先级低的类型转换为优先级高的类型。

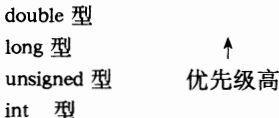


图4.2 数据类型的优先级

④在赋值表达式中,要把右边表达式的类型转换成左边表达式的类型,如练习4.11所示。

[练习4.11]

C)TYPE EXP4-11.C

```

main()
{
    int i, j;
    char c;
    float f;
    c = 50;
    printf("c = %c\t c = %d\n", c, c);
    i = 2.6; f = 2;
    printf("i = %d\t f = %f\n", i, f);
    i = 'a'; j = 'a' + 1;
    printf("i = %d\t j = %d\n", i, j);
    printf("c = %d\t c = %c\n", c, '0'c);
}

```

C)EXP4-11

```

c=2      c=50
i=2      f=2.000000
i=97     j=98
c=2      c=2

```

(2) 显式转换 一个表达式 exp 可用显式转换表达式

(type) exp

将其值转换为圆括号中 type 所指定的数据类型如：

```

int i = 10;
double f;
f = sqrt(double)i;

```

其中圆括号及其内的类型名即类型显式转换运算符。

读进 double 型数据，进而显示其值及截断后的值的程序，如练习 4.12 所示。

[练习 4.12]

C)TYPE EXP4-12C

```

main()
{
    double f;
    scanf("%lf", &f);
    printf("the float value %10.2f\n", f);
    printf("truncated is %d\n", (int)f);
}

```

C>exp4-12

```

12.2345678
the float value      12.23
truncated is 12

```

(3) 溢出问题 在进行运算时，时常会遇到结果产生溢出现象，如练习 4.13 所示。

[练习 4.13]

C)TYPE EXP4-13.C

```

main()
{
    int x, y, z;
    x = 400; y = 200;
    z = x * y;
    printf("z = %d\n", z);
}

```

C)EXP4-13

z=14464

在这个练习里，x、y、z 被定义为整型变量，x * y 的结果明显超出 int 型的数据范围。即使把变量 z 定义为 long 型，终因 x 和 y 的相乘是在整型范围内进行的，是不会有正确结果的。解决办法只有用类型显式转换，把 x、y 中的一个数转换为 long 型，这时，根据隐式转换规则，则另一个数也就自动转换为 long 型的了；然后再把 long 型的结果赋给事先已定义为 long 型的变量 z，如练习 4.14 所示。

C)TYPE EXP4-14.C

```

main()
{
    int x, y;
    long z;
    x = 400; y = 200;
    z = (long)x * y;
    printf("%ld", z);
}

```

C)EXP4-14

80000

6 运算符优先级

在 C 语言中，各运算符是有其运算的优先级别的。运算符的优先级决定了表达式的求解顺序。C 语言运算符的优先级如表 4.5 所示。

表 4.5 C 语言运算符优先级

运 算 符	分 类	结合性
() [] → .	单项运算符	从左至右
! ~ ++ -- (type) * & sizeof		从左至右
* / %		
+ -		
<< >>		
< (=) > =	双项运算符	从左至右
= = ! =		
&		
^		
&&	条件运算符	从右至左
?		
! =		
= + = - = * = / = % =	赋值运算符	从右至左
>> << & = ^ = =		
,	逗号运算符	从左至右

从该表我们可以看出：

(1) C 语言有 44 个运算符。

(2) 这些运算符分 15 个优先级别；优先级自上向下变低；其中 () 运算符优先级最高；逗号运算符优先级最低；赋值运算符的优先级也比较低。（转 22 页下）



智力竞赛抢答装置

宋济华

单片机是近几年来在国内计算机行业崛起的一颗新星。由于它具有体积小、价格低、编程效率高、控制能力强等特点,被广泛地应用在工业自动控制和仪器仪表智能化领域,是微机应用产品化的理想机型。本人利用 MCS-48 系列单片机 8035 制作了一台智力竞赛抢答装置。希望能给初学单片机的朋友以抛砖引玉的作用。

市售的智力竞赛抢答器大都是由分离式电子原件组装的。由于设计和布线困难,机器本身一般只具有抢答单一功能,举行智力竞赛还必须同时配备专门的计时器和计分器,机器的功能和可靠性都受到了极大的限制。本文介绍的智力竞赛抢答装置由于采用了先进的微电脑技术,使机器的性能更趋于完善。它集抢答器、计时器、计分器、音响提醒和数字显示于一体,具有较高的性能价格比,是举行智力竞赛的理想工具。

一、功能说明

本智力竞赛抢答装置(下文简称“装置”)由控制台、显示台和抢答器三部分组成。控制台设抢答、回答、复位、加分和减分五个按键式开关以及显示计时、抢答号和得分的六个数码显示器,是本装置的核心控制单元,显示台为八组共二十四个大型数码显示器,显示八个抢答组的当前得分。抢答台由八个按键式开关组成,参赛组各执一个。

装置上电后,初始化数据项并扫描控制台。当主持人出完题启动抢答键后,20 秒定时器开始计时并开放抢答台,计时显示器从 20 开始以秒为单位递减显示抢答剩余时间。为了渲染竞赛的紧张气氛,扬声器每秒都发出音长为 1/4 秒的短笛声。如果在 20 秒内无抢答,抢答号和计分显示器始终显示零,20 秒到后,扬声器发出三次“嘟嘟”声,以示抢答时间到,计时显示器回零且立即关闭抢答台,回到初始态。假如在 20 秒内有一组先抢答,扬声器中马上发出“叮咚”声,计时显示器跳回零,抢答号和计分显示器立即显示抢答组的组号和当前成绩。为了不致于其他组再抢答,有一组抢答后,装置立即关闭抢答台。此时,主持人可启动回答键,给回答者计时,计时显示器也从 20 开始以秒为递减单位显示回答剩余时间,不过此期间扬声器并不发出音响,目的是不致于影响观众听回答者的答题。20 秒后,扬声器发出三次“嘟嘟”报警声,以示回答时间到,计时显示器回零,主持人可根据回答的正误情况按动加分、减分键,并将此成绩计入到该组的总得分中,从显示器上反映出来。一旦按动了复位键,装置便回到初始态。

在智力竞赛中,参赛各组的基分数一般为 100 分,

以 10 为递增减单位。但在本装置中,为了简化电路和程序起见,以 10 为基分数,以 1 为递增减单位,而在显示器上反映出来的还是以 100 为基分数,10 为增减单位,这是因为在各组计分显示器的尾部多添了一个“0”。因此从效果上与前者是一致的。

二、硬件简介

本装置主要由单片机 8035、程序存储器 EPROM(2716)、地址锁存器 8212、数据输出锁存器(74LS273)、输入缓冲电路(74LS244)、显示驱动器、电源和外围器件组成。8035 内部集成了中央处理器(CPU)、数据存储器(RAM)、地址和数据总线(BUS)、定时器/计数器(CTC)以及两个并行的输入输出接口电路(P1 和 P2)是本装置的核心部件。2716 是本装置的 2K×8Bit 程序存储器;8212 是地址锁存器,由于 CPU 分时传输地址和数据信息,当 CPU 执行指令期间用它锁存地址的低八位信息;在显示电路中用的四片 8D 锁存器 74LS273,通过 8035P1 口提供的选片信号为后继的显示驱动器锁存 BUS 送来的段和位信息,两片 74LS244 是控制台和抢答台的输入缓冲电路接口;本装置中的六个显示驱动器为大型数码管显示器提供大功率的电流驱动。电源电路在本刊九〇年第 3 期《MCS48 交通信号自动控制》中已详细说明,不再赘述。

三、工作原理

图 1 是本装置的电原理图,图 2 是监控程序结构框图。当机器上电自举或复位后,8035 执行初始化,程序,P1 口各位依次变为低电平,为显示器数据锁存器提供选片信息,分时将各位数码显示器点亮并循环扫描控制键盘。当控制台有键闭合时,CPU 执行查询程序,译出键码,确定键意。若为抢答,监控启动 20 秒定时程序,并且每秒都在 P25 上输出一个高电平脉冲,触发短笛音电路。同时使 P17 变为低电平,选中抢答台键盘输入缓冲电路并循环扫描之。若在 20 秒定时期间,抢答台上有键闭合,监控立即使 P17 变为高电平,关闭抢答台,从而使以后的抢答失效,并在 P26 输出一个高电平脉冲,触发“叮咚”音响电路发声,尔后查询组号,并将抢答号和该组的成绩送控制台上的数码显示器且使计时器回零。定时到且无抢答,则 CPU 使 P27 输出高电平脉冲,触发“嘟嘟”电路发声三次,使 P17 为高电平关闭抢答台,计时器回零,操作返回初始态。

抢答台有抢答时,系统扫描控制台等待键盘输入,若回答键闭合,则监控重新启动 20 秒定时程序且等待控制台加分、减分或复位键闭合,以应答回答的正误,

并通过程序将加减的分数计入总成绩中显示出来,启动复位键,监控将操作返回初始态。

本装置的显示驱动器原理图见图 3,它是用晶体管和阻容元件放大数据锁存器后拖的 TTL 信号为大型数码管显示器的段信号和位信号提供足够功率的电流驱动。

本装置的监控程序的设计采用了“模块化”结构,在执行时采用子程序调用的方法来实现指定模块的功能。限于篇幅,监控程序表从略下面将几个主要模块的功能说明如下:

1. INT\$: 中断处理模块,它的作用是为装置提供精确的 20 秒定时。

2. DIGS: 显示模块,指令序列将 CPU 提供的显示

器段和位信息送显示器显示出来。

3. CZ: 拆字模块,是将计数值、抢答组号以及成绩值八位二进制数变成两位十进制数送显示模块 (DIGS)。

4. ZMB: 字模表模块,存储在 300~309 单元中的数字 0~9 的显示字模表。监控程序根据累加器 A 的值利用 MOV P3 A, @A 指令在程序存储器的第三页 (300~3FF) 单元内的 300~309 的字模信息。

5. DELAY: 延时 20ms 模块,配合键盘输入缓冲电路,该模块提供 20ms 延时,消除按键的抖动。

以上这些模块构成监控程序的基础,各种不同的控制功能都由这些模块堆积而成。

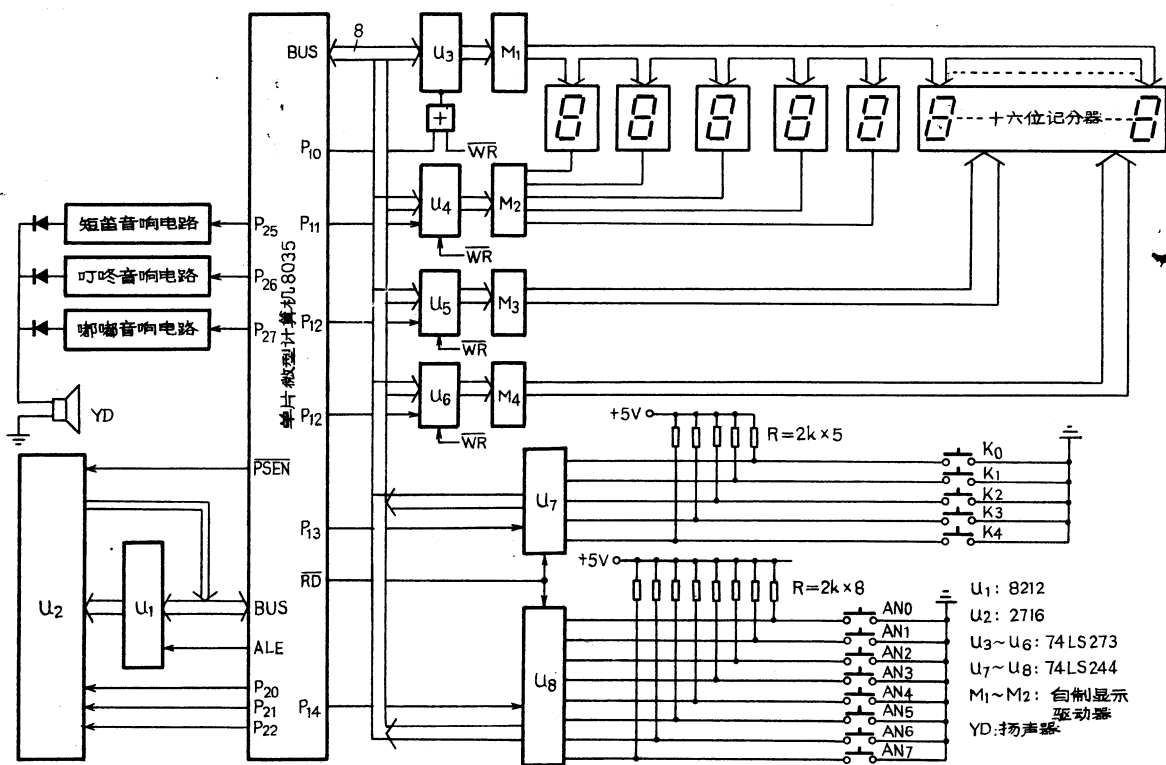


图 1

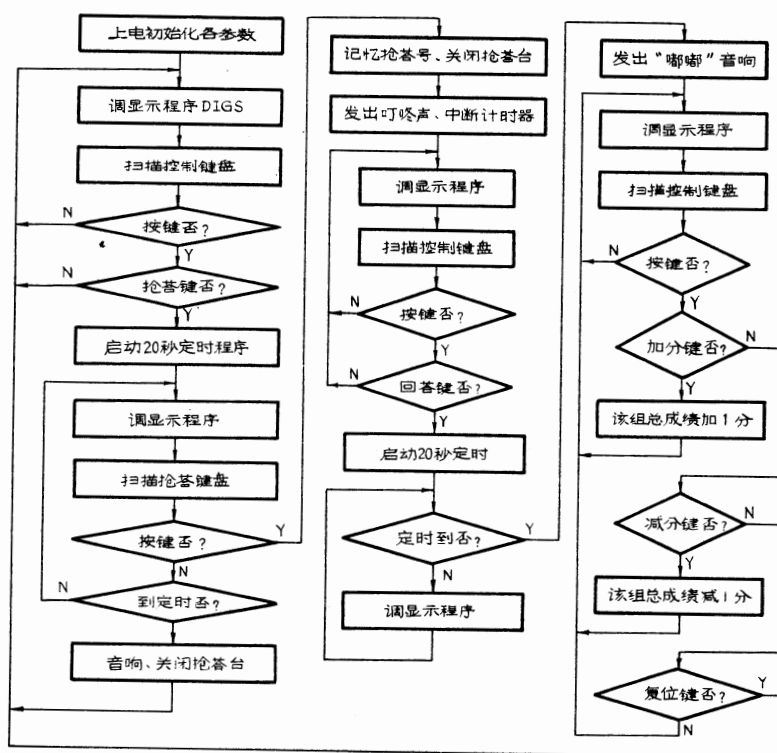


图 2

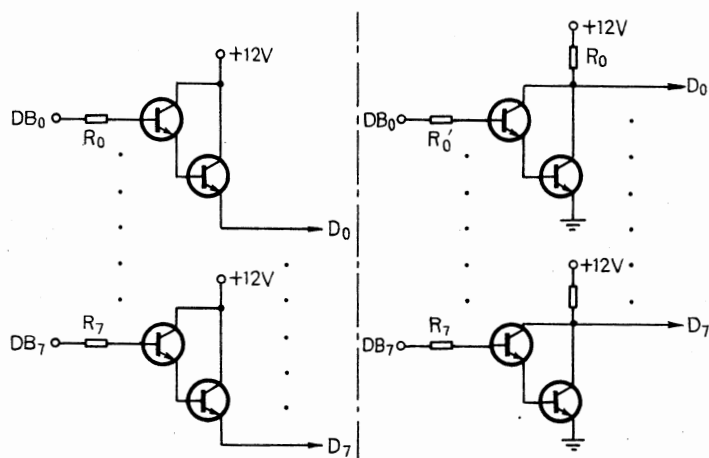


图 3

单片机电子音乐编程实践

张培仁 张新宇

音乐由一系列特定频率的声音组成。单片机能输出不同频率的方波信号形成音频,通过延时形成音长(节拍),这样,单片机就实现音乐演奏功能。

一、原理

通常采用十二平均律,将一个八度音程按十二半音频率按等比数列排列(见表一)。

对应着音名,可以很方便地写出各个音调音频的简谱符号。规则是:不论低中高音,1~2(1跟2),2~3,4~5,5~6,6~7,各相差一个音程,3~4,7~1各相差半个音程。例如G调的1的音频为196HZ,这一规律将用到程序设计之中。

为了用8051实现这些音频,我们用了定时器TO。定时器溢出后产生中断,将连接喇叭的输出口(P1.7)取反,即可产生方波形成音频。

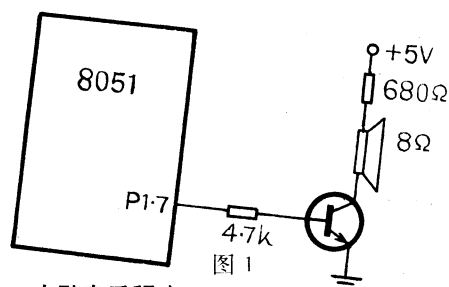
置TO为工作方式1,即TH0,TL0构成16位定时器,假定机器频率为0.5MHz,则对应于每个音频的时间常数见表一。

表一.时间常数(机器频率为0.5MHz)

NO. T	I	音名	音频 (HZ)	时间常数	简谱符号							
					C	D	E	F	G	A	B	
1	22	C	131	F88D	1							
2	23	C#	139	F8F8								
3	24	D	147	F95D	2	1						
4	25	D#	156	F9BC								
5	26	E	165	FA16	3	2	1					
6	27	F	175	FA6B	4			1				
7	28	F#	185	FABC		3	2					
8	29	G	196	FBO7	5	4			2	1		
9	30	G#	208	FB4F			3					
10	31	A	221	FB92	6	5	4	3	2	1		
11	32	A#	234	FBD2				4				
12	33	B	248	FC0E	7	6	5		3	2	1	
13	34	C	262	FC46	1			5	4			
14	35	C#	278	FC7C		7	6			3	2	
15	36	D	294	FCAE	2	1		6	5	4		
16	37	D#	312	FCDE			7				3	
17	38	E	330	FD0B	3	2	1	7	6	5	4	
18	39	F	350	FD35	4			1				
19	40	F#	371	FD5E		3	2		7	6	5	
20	41	G	393	FD83	5	4		2	1			
21	42	G#	416	FDA7		3			7	6		
22	43	A	441	FDC9	6	5	4	3	2	1		
23	44	A#	467	FDE9				4			7	
24	45	B	495	FE07	7	6	5		3	2	1	
25	46	C	525	FE23	1			5	4			

26	47	C#	556	FE3E		7	6			3	2	
27	48	D	589	FE57	2	1		6	5	4		
28	49	D#	624	FE6F			7				3	
29	50	E	661	FE85	3	2	1	7	6	5	4	
30	51	F	700	FE9A	4			1				
31	52	F#	742	FEAF		3	2		7	6	5	
32	53	G	786	FEC1	5	4		2	1			
33	54	G#	833	FED3			3			7	6	
34	55	A	882	FEE4	6	5	4	3	2	1		
35	56	A#	935	FEF4				4			7	
36	57	B	990	FF03	7	6	5		3	2	1	
37	58	C	1049	FF11				5	4			
38	59	C#	1112	FF1F		7	6			3	2	
39	60	D	1178	FF2B				6	5	4		
40	61	D#	1248	FF37			7				3	
41	62	E	1322	FF42					7	6	5	4
42	63	F	1400	FF4D								
43	64	F#	1484	FF57						7	6	5
44	65	G	1572	FF60								
45	66	A	1665	FF69							7	6
46	67	A#	1764	FF72								
47	68	B	1869	FF7A								7

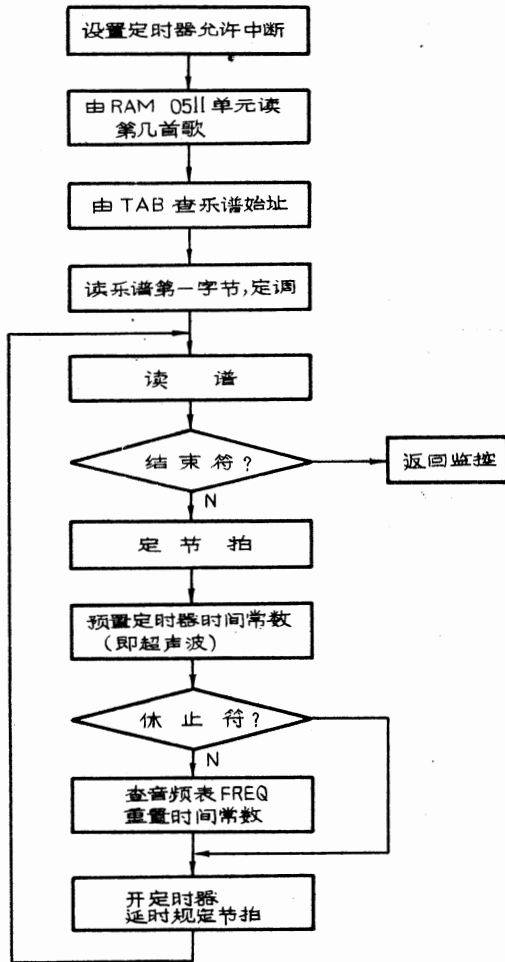
硬件配置见图1



二、自动奏乐程序

将编好的若干个乐曲程序,存放在EPROM中,即通过查表音符数据输出一段音乐。从乐曲程序中可以看出,此程序能表现C、D、E、F、G、A、B中的任意曲调,能表现从1到7的二十一个音频、休止符和十种节拍。最关键是的,采用压缩数据,一个字节即包含音频和音长两个信息,节省很大内存。

程序结构框图如图二。



图二

自动奏乐程序清单

```

1  ORG 2000H
2  SPK EQU P1.7
3  MOV IE, #82H          ;允许中断
4  ANL 89H, #0F0H        ;置定时器为工作方式 1
5  ORL 89H, #01H
6  AJMP START
7  ORG 200BH              ;中断服务程序
8  CPL SPK                ;取反,产生方波
9  MOV TH0, R7
10 MOV TL0, 6R
11 STEB TR0
12 RETI
13 START: MOV R5, 05H      ;选歌
14 star: MOV DPTR, #TAB    查乐谱首址
15 MOV A, R5
16 DEC A
17 RL A

```

```

18  MOVC A, @A+DPTR
19  PUSH A
20  MOV A, R5
21  DEC A
22  RL A
23  INC A
24  MOVC A, @A+DPTR
25  PUSH A
26  POP DPL
27  POP DPH
28  MOV A, #00H           ;定调
29  MOVC A, @A+DPTR
30  MOV R3, A
31  INC DPTR
32  SONG: CLR SPK
33  MOV A, #00H           ;读乐谱
34  MOVC A, @A+DPTR
35  MOV R2, A
36  SWAP A
37  ANL A, #0FH
38  JZ EXIT                ;结束符,转 EXIT
39  MOV B, #05H
40  DIV AB
41  XCH A, B
42  JNZ MOM
43  MOV A, #05H
44  DEC B
45  MOM: MOV R0, A
46  MOV R1, B
47  MOV A, R2
48  ANL A, #0FH
49  MOV B, #08H
50  DIV AB
51  MOV R2, B
52  MOV R4, A
53  MOV R7, #0FFH         ;预置为休止符
54  MOV R6, #0F9H
55  MOV A, R2
56  JZ YANS                ;休止符则转
57  CLR C                  ;转化为余数处理数
58  RLC A                  ;余数:1234567
59  DEC A                  ;余数处理数:13568AC
60  MOV R2, A
61  CLR C
62  SUBB A, #06H
63  JC AN
64  DEC R2
65  AN: MOV A, #0CH
66  MOV B, R1
67  MUL AB

```

68	ADD A,R2		115	DB 0FDH,5EH,0FDH,83H,0FDH,0A7H,0FDH,0C9H,0FDH,0E9H,0FEH,07H
69	YANS;PUSH DPH		116	DB 0FEH,23H,0FEH,3EH,0FEH,57H,0FEH,6FH,0FEH,85H,0FEH,9AH
70	PUSH DPL		117	DB 0FEH,0AFH,0FEH,0C1H,0FEH,0D3H,0FEH,0E4H,0FEH,0F4H,0FFH,03H
71	JZ TM	:休止则转 TM	118	DB 0FFH,11H,0FFH,1FH,0FFH,2BH,0FFH,37H,0FFH,42H,0FFH,4DH
72	DEC A	:否则,查音频表 FREQ	119	DB 0FFH,57H,0FFH,60H,0FFH,69H,0FFH,72H,0FFH,74H,0FFH,81H
73	ADD A,R3	:重置时间常数	120	LONG;DB 08H,20H,30H,80H,0C0H ;延时常数,代表 1/4,1,3/2,4,6 拍
74	RL A		121	TAB;DB 21H,22H,21H,47H,21H,81H,21H,0B7H,21H,0DFH,22H,46H,22H,72H
75	MOV DPTR,#FREQ		122	ORG 2122H
76	MOV R1,A		123	T1;DB 09H ;A 小调 4/4 日本歌曲
77	MOVC A,@A+DPTR		124	DB 73H,7BH,7AH,71H,79H,2FH,36H,26H,4EH,四季歌
78	MOV R7,A		125	DB 74H,7CH,7BH,7AH,79H,7AH,7CH,93H
79	MOV A,R1		126	DB 74H,7CH,7BH,72H,7AH,7CH,73H,7BH,79H,26H,26H
80	INC A		127	DB 27H,73H,7AH,79H,2FH,79H,46H,00H
81	MOVC A,@A+DPTR		128	ORG 2147H
82	MOV R6,A		129	T2;DB 05H ;F 调 2/4 苏格兰民歌
83	TM;MOV TH0,R7		130	DB 2DH,89H,69H,79H,7BH,7AH,69H,72H,73H,;友谊地久天长
84	MOV TL0,R6		131	DB 89H,69H,7BH,7DH
85	SETB TR0	:开定时器	132	DB 76H,7EH,8BH,6BH,7BH,79H
86	MOV DPTR,#LONG	:查延时常数	133	DB 8AH,69H,7AH,7BH,89H,6EH,2EH,2DH
87	MOV A,R0		134	DB 8EH,7EH,8DH,6BH,7BH,79H
88	DEC A		135	DB 8AH,69H,7AH,7EH,8DH,6BH,7BH,7DH
89	MOVC A,@A+DPTR		136	DB 86H,0C9H,8DH,6BH,7BH,79H
90	MOV R0,A		137	DB 8AH,69H,7AH,7BH,89H,1EH,2EH,2DH,81H,00H
91	MOV A,R4		138	ORG 218H
92	JZ TTT	:使用节拍 1	139	T3;DB 09H ;A 调 4/4 日本民歌
93	MOV A,R0	:使用节拍 2	140	DB 73H,73H,76H,77H,0C1AH,77H,9EH,;荒城之月
94	CLR C		141	DB 74H,74H,73H,72H,0ABH,70H
95	RRC A		142	DB 73H,73H,76H,77H,0C1H,77H,9EH
96	MOV R0,A		143	DB 74H,74H,73H,73H,5EH,70H
97	TTT;ACALL DELAY	:延时指定节拍	144	DB 81H,69H,27H,26H,74H,74H,9BH
98	DJNZ R0,TTT		145	DB 72H,73H,74H,74H,0ABH,70H
99	POP DPL		146	DB 73H,73H,76H,77H,0CH1,77H,9EH
100	POP DPH		147	DB 74H,74H,73H,73H,5EH,70H,00H
101	INC DPTR		148	ORG 21B7H
102	AJMP SONG	:继续演奏	149	T4;DB 07H ;G 调 4/4 日本歌曲
103	EXIT;CLR SPK	:退出	150	DB 81H,7BH,7AH,79H,2FH,79H,72H,72H,9AH,;你叫什么名字?
104	CLR TR0		151	DB 82H,7CH,7BH,7AH,79H,7AH,73H,9BH
105	LJMP 0000H			
106	DELAY;MOV R4,#40H	:延时子程序		
107	LOOP1;MOV R1,#35H			
108	LOOP2;NOP			
109	DJNZ R1,LOOP2			
110	DJNZ R4,LOOP1			
111	RET			
112	FREQ;DB,0F8H,8DH,0F8H,0F8H,0F9H,5DH,0F9H,0BCH,0FAH,16H,0FAH,6BH			
113	DB,0FAH,0BCH,0FBH,07H,0FBH,07H,0FBH,4FH,0FBH,92H,0FBH,0D2H,0FCH,0EH			
114	DB 0FCH,46H,0FCH,7CH,0FCH,0AEH,0FCH,0DEH,0FDH,0BH,0FDH,35H			

152 DB 85H,7EH,7DH,7BH,7CH,7DH,76H,76H,9EH
153 DB 85H,7CH,7BH,7BH,7AH,7AH,79H,79H,79H,
79H,99H,00H

154 ORG 21DFH

155 T5:DB 07H ;G 调 4/4 美国歌曲

156 DB 7DH,0CBH,0CAH,0C9H,85H,7DH,7DH
铃儿响叮当

157 DB 7DH,0CBH,0CAH,0C9H,86H,7EH

158 DB 7EH,0CCH,0CBH,0CAH,87H,7FH

159 DB 0CDH,0CDH,0CCH,0CAH,0D3H,0C9H

160 DB 7DH,0CBH,0CAH,0C9H,85H,7DH

161 DB 7DH,0CBH,0CAH,0C9H,86H,7EH

162 DB 7EH,0CCH,0CBH,0CAH,0CDH,0CDH,0CDH,
0CDH

163 DB 0CEH,0CDH,0CCH,0CAH,0D1H,28H

164 DB 0CBH,0CBH,0C3H,0CBH,0CBH,0C3H

165 DB 0CBH,0CDH,0D9H,0BAH,0EBH

166 DB 0CCH,0CCH,0DCH,0BCH,0CCH,0CBH,0CBH,
0CBH

167 DB 0CBH,0CAH,0CAH,0C9H,0C2H,0C5H

168 DB 0CBH,0CBH,0C3H,0CBH,0CBH,0C3H

169 DB 0CBH,0CDH,0D9H,0BAH,0EBH

170 DB 0CCH,0CCH,0DCH,0BCH,0CCH,0CBH,,0CBH,
0CBH

171 DB 0CDH,0CDH,0CCH,0CAH,0D1H,28H,00H

172 ORG 2246H

173 T6:DB 09H ;E 调 2/4《桥》插曲

174 DB 2BH,2EH,2FH,79H,4EH ;啊,朋友再见

175 DB 2BH,2EH,2FH,79H,4EH

176 DB 2BH,2EH,2FH,71H,2FH,2EH

177 DB 71H,2FH,2EH,73H,73H,73H,7AH,7BH

178 DB 7CH,9CH,7EH,7DH,7CH,7EH,9BH

179 DB 7BH,7AH,79H,27H,73H,71H,27H

180 DB 3EH,30H,3EH,28H,00H

181 ORG 2272H

182 T7:DB 04H ;E 调 3/4 爱的罗曼史

183 DB 0C3H,0C3H,0C3H,0C3H,0C2H,0C1H

184 DB 0C1H,77H,76H,76H,0C1H,0C3H

185 DB 0C6H,0C6H,0C6H,0C6H,0C5H,0C4H

186 DB 0C4H,0C3H,0C2H,0C2H,0C3H,0C4H

187 DB 0C3H,0C4H,0C3H,0C5H,0C4H,0C3H

188 DB 0C3H,0C2H,0C1H,0C1H,77H,76H

189 DB 77H,77H,77H,77H,0C1H,77H

190 DB 76H,76H,76H,00H

1. 选歌

在通用程序中可以输入多首乐曲(以 0×H 结尾),而将每首乐曲乐谱首地址按顺序放入表 TAB 中(见程序清单)。我们从 RAM 某个单元,此为 05H 单元读出第几首歌,查表 TAB 即可得到乐曲乐谱首址。演

奏乐曲直到结束字 0×H。本程序写有七首歌。

2. 选调

乐曲乐谱表第一字节是用来定调的(存入 R₃)。也就是在查音频表 FREQ 时须加上的偏移量。基准是 C 调,即偏移量为 0(见表二)。其它音调各简谱符号相对于 C 调都有一个偏移量,D 调偏移量是 2,故用 02H 来定 D 调。

表二:

音调	C	D	E	F	G	A	B
偏移量	0	2	4	5	7	9	B

3. 音频和音长(节拍)

我们设计了一种算法,使压缩数据能表现低中高音,不同的音长、休止符和结束符。

乐曲中一个字节为一个单位,即可表现音频和音长。字节高四位取自表三,低四位取自表四。高四位数包含音长和音高(即是低音,中音还是高音)两个信息。音长在高四位数中尚不明确,使用拍节 1 还是拍节 2 还需由低四位数肯定。高四位为 0 表示结束符。低四位指出使用的节拍和音符。不论是 1、1 或 1,低四位都应是 1 或 9。低四位为 0 或 8 表示休止符。具体是:高四位数除以 5 的余数指出拍节 1 我们的算法假使 5 除以 5 得 0 余 5,10 除以 5 得 1 余 5……),查表 LONG(见程序清单)得延时常数。然后低四位数除以 8,如商为 0 表示延时常数不变,即拍节 1。否则,延时常数除以 2,即拍节 2。拍节 2 是拍节 1 的一半。

表三. 高 4 位值

拍节 1	1/4	1	3/2	4	6
拍节 2	1/8	1/2	3/4	2	3
低 音	1	2	3	4	5
中 音	6	7	8	9	A
高 音	B	C	D	E	F

表四. 低 4 位值

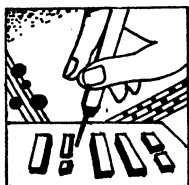
音符	1	2	3	4	5	6	7	0
拍节 1	1	2	3	4	5	6	7	0
拍节 2	9	A	B	C	D	E	F	8

LONG 中的延时常数分别代表 1/4、1、3/2、6 拍,延时间可由程序延时部分(标号 TTT)算出。实际上还应加上数据处理过程的时间,所以实际延时比表五中的长。

表五. 延时时间

拍节	1/8	1/4	1/2	3/4	1	3/2	2	3	4	6
延时(s)	0.080	0.160	0.330	0.490	0.660	0.981	1.311	1.972	2.623	3.93

(转至 39 页下)



学装微电脑

数字游标长尺的制作

易齐干

在微电脑控制技术中,往往要测量物体的长度,这时,必须把测得的尺寸变换为微电脑能理解的数字量。

本文以数字游标长尺为例,介绍微电脑测长度的原理,它是利用滑动电位器,将长度转换为电压,再用模拟量/数字量(A/D)变换电路将电压变换为8位数据,通过UP-80显示其测量值,亦用打印机打印出结果。

1. 数字游标卡尺的结构

数字游标卡尺如图1所示。由两部份组成:一是滑动电位器为主体的机械部份,它把长度尺寸转换为电压。另一部份是接口,它把电压变换为8位数字量。

该数字游标卡尺的滑动式电位器可变范围最大为40mm,所以改变定程档块的位置,能测量的范围分别为:0~40mm;20~60mm;40~80mm;60~100mm;80~120。

A/D变换电路TC5090P可将滑动式电位器测得的电压变换为8位数字量。它是将0~5V模拟电压变换为00H~FFH的8位数字量的IC。分辨率为256(电压为5/256V分辨率),即,0.5mm~52AH对应;0.0mm~528H对应;400mm~SC8H对应。这样,若在00H~FFH范围内选择较为合适。不按上述原则也可以。

尺寸测量值、8位数据与电压的关系如表1所示。

表1 测量值和8位数据与电压的关系

测量值(mm)	0.0	0.5	1.0	39.0	39.5	40.0
8位数据 (16进制)	28H	2AH	2CH	C4H	C6H	C8H
电压(V)	0.78	0.82	0.86	3.83	3.87	3.91

注1)测量值为0.0mm时电压为 $(16 \times 2 + 8) \times 5 / 256 \text{ V} = 0.78 \text{ V}$

注2)每0.5mm的电压变化为 $5 / 256 \times 2 = 0.04 \text{ V}$

表2 STC,RSEL,L,H信号工作关系

工作端	STC	RSEL
A/D转换开始	H	H
要求高位数据	L	H
要求低位数据	L	L

例如,滑动式电位器得到的电压为3.87V,TC5090P输出数据C6H,将它输入给微电脑,则数字显示为39.5mm。

2. A/B转换电路(图2)

有各种各样的方法可把模拟量转换为数字量。本文使用东芝制造的TC5090P据有A/D转换功能。来自微电脑的低电平(0V)或高电平(5V)分别出现在STC端或RSEL端,或开始A/D转换,或将转换后的8位数据分为高4位、低4位,由D₄~D₁端输出。

STC、RSEL高电平、低电平与工作关系如表2所示。EOC端为低电平时,该IC正在进行A/D转换;EOC为高电平时,表示转换结束。

图2所示的电路中,有两片型号为74LS157,其作用是将较多的数据输入给具有较少输入端的微电脑。

这两片IC各具备四位的输入端口A、B,以SEL端为低电平、高电平决定/IC的输出,SEL的高电平、低电平与信息流动如表3所示。

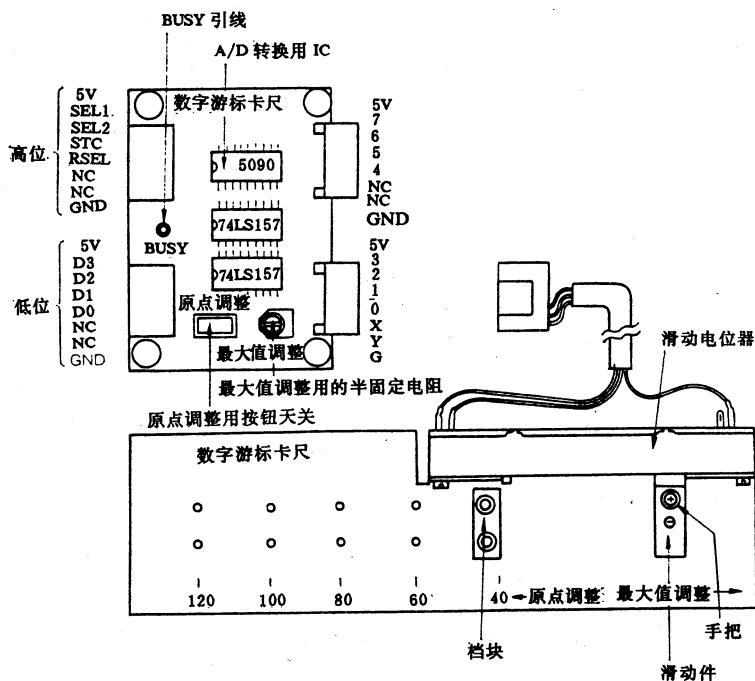
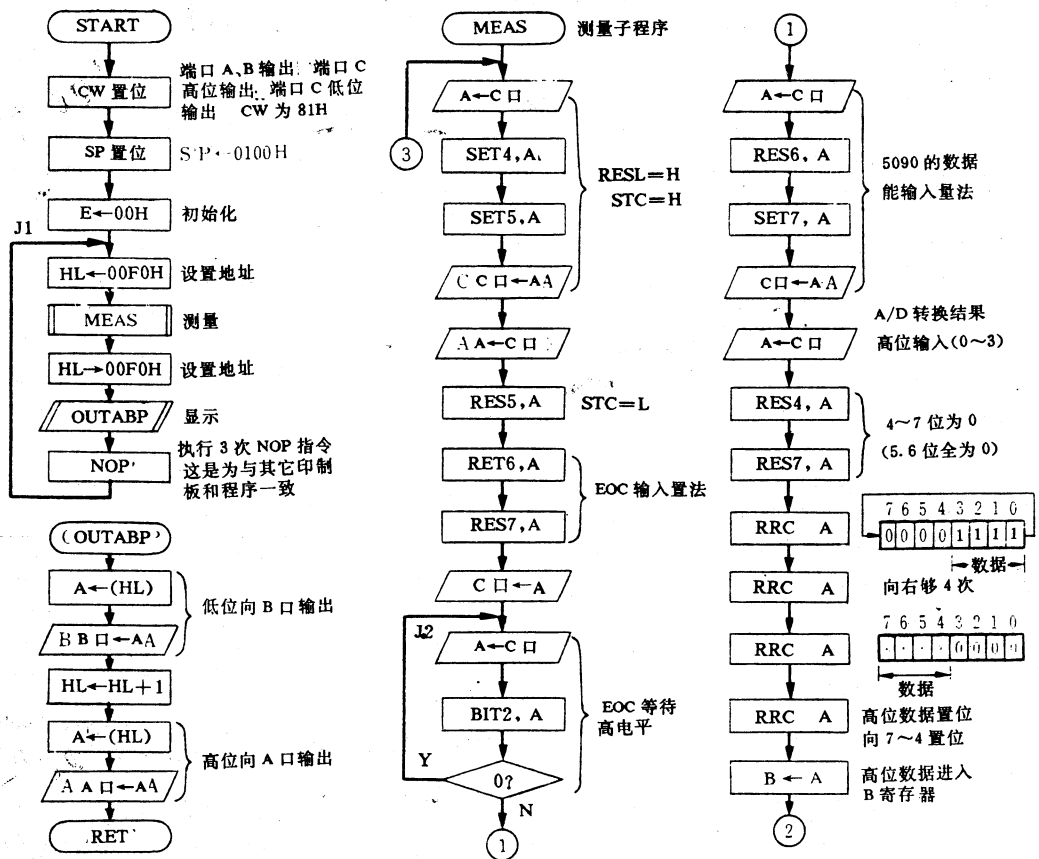


图1

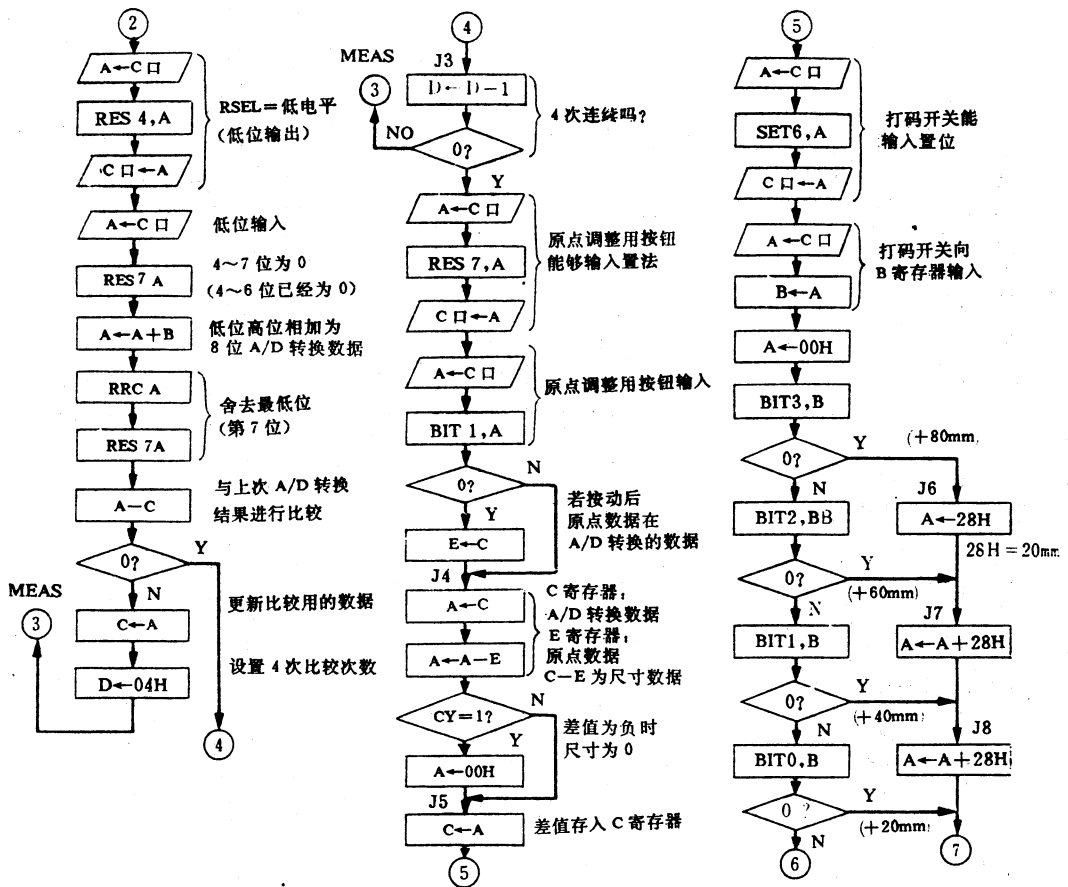
表 4. (1)



标号	助记符	地址	机器语	注释
START	LD A, 81H	0000	3E 81	输入、输出置位
	OUT (03H), A	0002	D3 03	
	LD SP, 0100H	0004	31 00 01	SP ← 0100H
	LD E, 00H	0007	1E 00	初始化
J1	LD HL, 00F0H	0009	21 F0 00	设置地址
	CALL MEAS	000C	CD 1B 00	测量子程序
	LD HL, 00F0H	000F	21 F0 00	设置地址
	CALL OUTABP	0012	CD E6 00	显示
	NOP	0015	00	
	NOP	0016	00	什么也不做
	NOP	0017	00	
	JP J1	0018	C3 09 00	
MEAS	IN A, (02H)	001B	DB 02	
	SET 4, A	001D	CB E7	RESL = H
	SET 5, A	001F	CB EF	STC = H
	OUT (02H), A	0021	D3 02	
	IN A, (02H)	0023	DB 02	
	RES 5, A	0025	CB AF	STC = L
	RES 6, A	0027	CB B7	
	RES 7, A	0029	CB BF	EOC 输入上置位
	OUT (02H), A	002B	D3 02	

标号	助记符	地址	机器语	注释
J2	IN A, (02H)	002D	DB 02	
	BIT 2, A	002F	CB 57	EOC = 高电平
	JP Z, J2	0031	CA 2D 00	
	IN A, (02H)	0034	DB 02	
	RES 6, A	0036	CB B7	5090 的数据纯输入置位
	SET 7, A	0038	CB FF	
	OUT (02H), A	003A	D3 02	
	IN A, (02H)	003C	DB 02	高位输入
	RES 4, A	003E	CB A7	4~7 位为 0
	RES 7, A	0040	CB BF	
	RRC A	0042	CB 0F	
	RRC A	0044	CB 0F	高位数据向 4~7 位置位
	RRC A	0046	CB 0F	
	RRC A	0048	CB 0F	
	LD B, A	004A	47	
	IN A, (02H)	004B	DB 02	
	RES 4, A	004D	CB A7	RESL = 低电平
	OUT (02H), A	004F	D3 02	
	IN A, (02H)	0051	DB 02	输入低位 4~7 位为 0
	RES 7, A	0053	CB BF	定 8 位数据
	ADD A, B	0055	80	

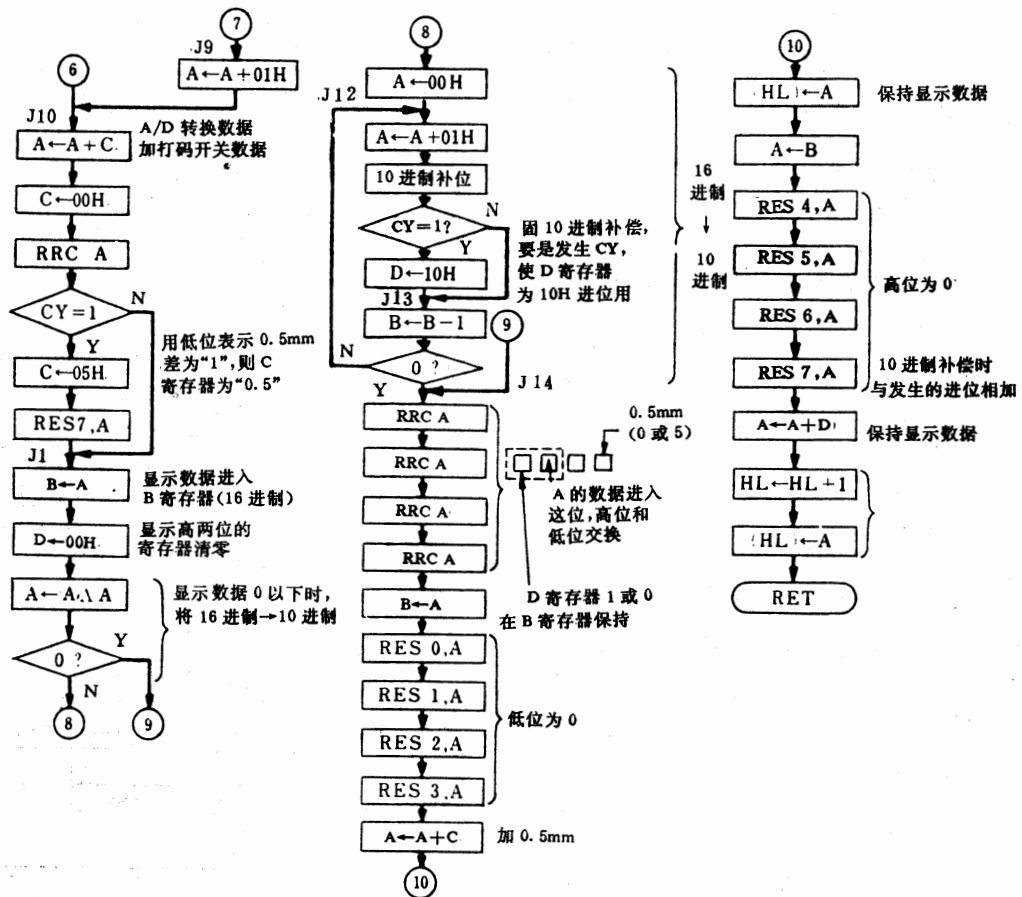
表 4. (2)



标号	助记符	地址	机器语	注释
J3	RRC A	0056	CB 0F	舍掉最低位
	RES 7, A	0058	CB BF	
	CP C	005A	B9	比较前次数据
	JP Z, J3	005B	CA 64 00	
	LD C, A	005E	4F	更新比较数据
	LD D, 04H	005F	16 04	比较次数置位
	JP MEAS	0061	C3 1B 00	
	DEC D	0064	15	
	JP NZ, MEAS	0065	C2 1B 00	连续 4 次吗?
	IN A, (02H)	0068	DB 02	原点调整用按钮
J4	RES 7, A	006A	CB BF	开关输入置位
	OUT (02H), A	006C	D3 02	
	IN A, (02H)	006E	DB 02	按钮开关
	BIT 1, A	0070	CB 4F	输入检验
	JP NZ, J4	0072	C2 76 00	
	LD E, C	0075	59	原点数据
	LD A, C	0076	79	尺寸数据
	SUB E	0077	93	
	JP NC, J5	0078	D2 7C 00	负号尺寸为 0
	XOR A	007B	AF	
J5	LD C, A	007C	4F	

标号	助记符	地址	机器语	注释
J5	IN A, (02H)	007D	DB 02	打码开关
	SET 6, A	007F	CB F7	输入置位
	OUT (02H), A	0081	D3 02	
	IN A, (02H)	0083	DB 02	
	LD B, A	0085	47	
	XOR A	0086	AF	
	BIT 3, B	0087	CB 58	检验打码
	JP Z, J6	0089	CA 9E 00	开关 S ₃
	BIT 2, B	008C	CB 50	检验打码
	JP Z, J7	008E	CA A0 00	开关 S ₂
J6	BIT 1, B	0091	CB 48	检验打码
	JP Z, J8	0093	CA A2 00	开关 S ₁
	BIT 0, B	0096	CB 40	检验打码
	JP Z, J9	0098	CA A4 00	开关 S ₀
	JP J10	009B	C3 A6 00	
	LD A, 28H	009E	3E 28	+80mm
	ADD A, 28H	00A0	C6 28	+60mm
	ADD A, 28H	00A2	C6 28	+40mm
	ADD A, 28H	00A4	C6 28	+20mm
	ADD A, C	00A6	81	A/D 转换数据加
J10	LD C, 00H	00A7	0E 00	打码开关数据

表 4. (3)



标号	助记符	地址	机器语	注释
J11	RRC A	00A9	CB 0F	最低位为 0.5 (保持显示数据) (16 进制)
	JP NC, J11	00AB	D2 B2 00	
	LD C, 05H	00AE	0E 05	
	RES 7, A	00B0	CB BF	
	LD B, A	00B2	47	
J12	LD D, 00H	00B3	16 00	显示数据为 0 时 16 进制→10 进制
	AND A	00B5	A7	
	JP Z, J14	00B6	CA C6 00	
	XOR A	00B9	AF	
	ADD A, 01H	00BA	C6 01	
J13	DAA	00BC	27	16 进制→10 进制
	JP NC, J13	00BD	D2 C2 00	
	LD D, 10H	00C0	16 10	
	DEC B	00C2	05	
	JP NZ, J12	00C3	C2 BA 00	
J14	RRC A	00C6	CB 0F	高位与低位交换
	RRC A	00C8	CB 0F	
	RRC A	00CA	CB 0F	
	RRC A	00CC	CB 0F	
	LD B, A	00CE	47	
	RES 0, A	00CF	CB 87	低位为 0

标号	助记符	地址	机器语	注释
J14	RES 1, A	00D1	CB 8F	低位为 0
	RES 2, A	00D3	CB 97	
	RES 3, A	00D5	CB 9F	
	ADD A, C	00D7	81	0.5mm 数据相加 保持显示数据
	LD (HL), A	00D8	77	
	LD A, B	00D9	78	
	RES 4, A	00DA	CB A7	高位为 0
	RES 5, A	00DC	CB AF	
	RES 6, A	00DE	CB B7	
	RES 7, A	00E0	CB BF	加进位 保持显示数据
	ADD A, D	00E2	82	
	INC HL	00E3	23	
	LD (HL), A	00E4	77	B 端口 输出低两位 端口 A 输出高两位
	RET	00E5	C9	
	LD A, (HL)	00E6	7E	
	OUT (01H), A	00E7	D3 01	
	INC HL	00E9	23	
OUT ABP	LD A, (HL)	00EA	7E	
	OUT (00H), A	00EB	D3 00	
	RET	00ED	C9	

现在讨论将 UP-80 与数字游标卡尺连接起来 3), 对其测量尺寸进行数字显示的方法。

	PC ₆ (SEL1)	PC ₇ (SEL2)
TC5090P 的 D ₄ ~ D ₁₁	L	H
打码开关	H	低电平高电平均可
TC5090 的 EOC 按键开关 BUSY	L	L

上述程序中, MEAS 子程序有它较方便的特点。即初始化将 00H 置于 E 寄存器, 调用子程序之前, 将存放 A/D 转换结果的地址置于 HL 寄存器时, 测量值自动地进入 HL 寄存器对所指定的两个地址。例如, HL 寄存器对指定 00F0H 地址, 测量结果 10.25mm 时, 00F0H 地址低位存 25, 00F1H 地址高位存 10。

(1) 数字游标卡尺原点调整与最大值调整这里讨论 0~40mm 测量范围的操作方法。此时, 打码开关部件的打码开关全部为不接通状态。测量范围改变后, 要接通打码开关部件, 请参考表 5 并按下述要点进行。

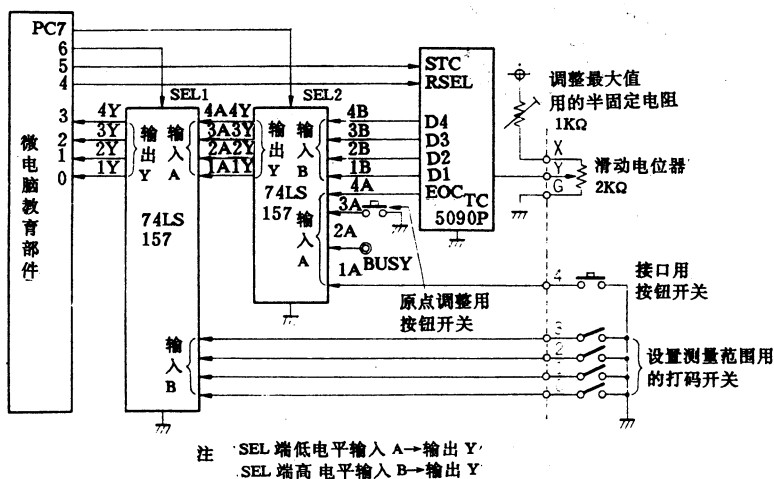
②照图 4 程序清单顺序写入程序

④若写入完毕,将写入控制部件的肘式开关①至读出侧,检查写入的程序是否有差错。

⑥按动中央处理部件的复位开关,时间为 1 秒左右。将位于写入。读出侧的肘式开关④搬至实行侧。

⑦搬动滑动电位器的滑动件,将它移动到原点调整侧。

⑧按动接口印刷线路板上的原点调整开关,于是,4位数字显示组件显示 $\overline{0000}$ 。



—

0	4	0	0
---	---	---	---

 表示 40.0mm.

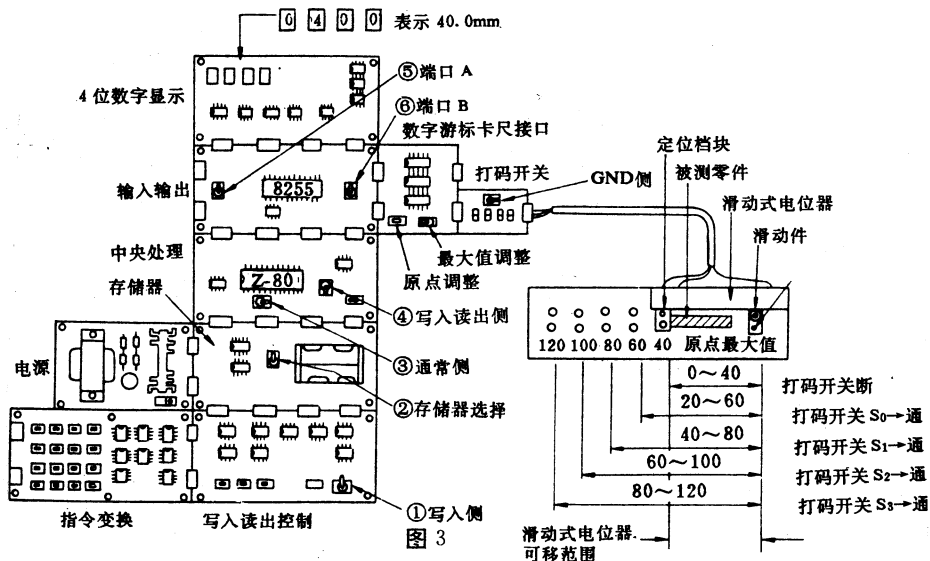


表 5 测量范围与打码开关的关系

测量范围 (mm)	打码开关的状态 (0→通 X→断)				挡块的位置	原点调整 显示	量最大值 调整显示
	S ₃	S ₂	S ₁	S ₀			
0~40	×	×	×	×	40	0000	0400
20~60	×	×	×	○	60	0200	0600
40~80	×	×	○	×	80	0400	0800
60~100	×	○	×	×	100	0600	1000
80~120	○	×	×	×	120	0800	1200

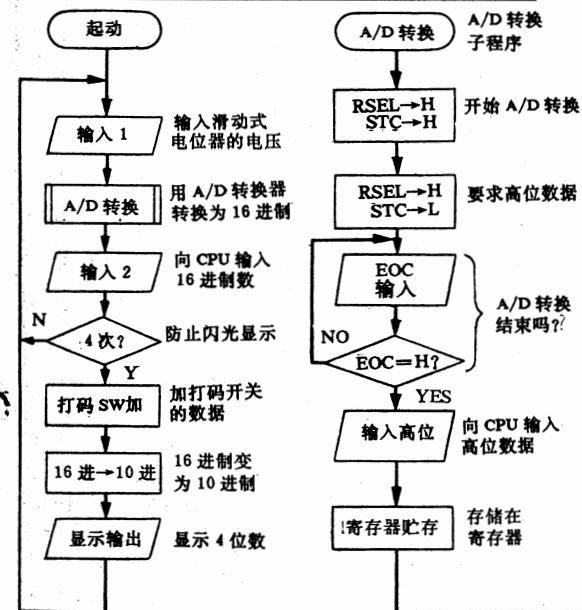


图 4 简单的流程图

⑨搬动滑动电位器的滑动件,将它移动到最大值调整侧。

⑩调整接口印刷电路板上的调整最大值半固定电

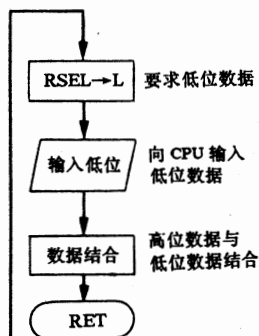
(上接 33 页) 高四位数除以 5 的商乘以 12,加上低四位数除以 8 的余数的处理数,再加上音调偏移量,即可以从音频表 FREQ 中查出时间常数,送定时器产生音乐。需要说明的是,表一中每个音符是分散的。为了查出同一调下各音符的时间常数,需要把低四位除以 8 的余数转换为查表值,暂称为余数处理数。它等于同一调下,同一音高中在表 1 中的位置。譬如下调的 5,余数处理数是 8,显然,不论什么调,不论音高,“哆”音的余数处理数均为 1,“咪”的均为 3,等等。

如果低四位数除以 8 的余数为 0,则表示休止符,喇叭不响。为表现得更自然,不采用关定时器的方法。而是预置时间常数为 FFF9H。如为休止符,时间常数不变,产生超声波。否则根据音频表,产生音乐。

4. 停止

表 6 开关的关系

部件名称	开关 No	开关方向
写入读出控制	①	读出侧
存储器	②	存储器选择
中央处理	③	通常测
	④	写入、读出侧
输入输出	⑤	端口 A 侧
	⑥	端口 B 侧



阻,直至 4 位数字显示部件显示出 H 0 4 0 0 为止。

(2)测量

①如图 3 所示,将被测零件与挡块重址放置。

②松动手柄,将滑动电位器的滑动件朝被测零件方向慢慢移动

③滑动件接触被测另件时,4 位数字显示部件显示测量尺寸。

该数字游标卡尺分辨率为 0.5mm,略为粗糙一些。不加改进将它应用于实际测量存在一定问题。但是,可以认为对于我们学习把测量尺寸输入给微电脑,进而数字显示还是

具有充分肯定的价值。

要提高数字游标卡尺的分辨能力,一是要求滑动电位器具有较高的可靠性,二是要提高电源的稳定程度。对此有兴趣的读者不妨请进行制作尝试。

高四位为 0 表示停止。程序指向 0000H 如有监控则返加监控,重置 RAM05 单元,奏另外的歌曲。事实上,很简单的改动即可使程序连奏成重奏。比如,此时的低四位为 0 表示退出,非 0 表示重奏次数,为 F 表示连奏(奏下一首歌)。

5. 如何编写乐谱表

(1)乐谱表首址存放在表 TAB 中

(2)定调。由表二查出偏移量作为第一个字节。

(3)由表三、表四将乐谱译成数据。举例说明:

| 35 1·2 3 · 0 |

其中的“1”为中音,节拍 3/4,由表三查出高四位为“8”;使用节拍 2,从表四查出低四位数为“9”,故为 89H。整段乐谱数据是:7BH,7DH,89H,62H,83H,28H。

(4)结尾必须是 0×H。

初级程序员水平考试辅导讲座

微机数据录入操作问答

刘英 张茂芝

1. 微机数据录入操作的含义是什么?

数据录入的全称是计算机辅助数据录入(CADE——Computer Aided Data Entry),又称数据转换(data conversion)。数据是信息的载体。由于常见的数据是文字、数字、图表、符号等,所以用计算机处理数据就必须进行数据录入——将数据转换成机器的可读形式。这里数据录入与传统打字不完全相同,它必须按照一定的要求,遵循某种约定(如格式的约定、控制字约定等,)借助于计算机进行操作。

随着计算机技术的迅速发展,数据录入越来越显得重要,从六十年代开始数据录入迅速发展,至今已形成一个数据录入行业,为建立和更新数据库、办公室自动化管理、出版印刷业的电子排版印刷等方面服务。

数据录入常用设备主要有:数据录入专用设备、微型计算机、中英文电子打字机等。作为初级程序员主要指能在微机上(如 IBM PC 及其兼容机)进行数据录入操作。

2. 数据录入操作的基本要求是什么?

数据录入操作的基本要求是完成用户对作业(录入任务称为作业)的要求,保证录入数据的正确性。

为此要求录入人员具有很好的素质。例如,具有一定的中文水平和英文阅读能力;了解计算机的软硬件基本知识,能熟练地进行键盘操作;能够识别中英文书写体;熟悉校对符号和排版规则;能熟练使用常用的录入软件;以及具有紧张、细致、认真、负责的工作态度等。

3. 数据录入操作的要领主要有哪些?

数据录入操作是一门技术。掌握这门技术需要注意下面几点。

(1)培养对键盘录入的兴趣,树立不怕艰苦的工作作风。数据录入这项工作在初学时,难免感到单调乏味,可惜又无捷径可循。所以要培养对此项工作的兴趣,精力集中,不怕苦,从易到难,由简入繁,持之以恒,循序渐进地按正确的操作方法和顺序坚持练下去。当能独立上岗进行录入工作时,要不断总结经验,提高录入的速度和质量。

(2)保持良好的录入姿势,减轻工作的疲劳程度。在进行数据录入时,录入员的较好姿势应该是平坐在

高度合适的椅子上,两腿平放在桌下,光线从左面过来。录入材料最好放在带有滑尺的架子上。录入员两肘悬空手腕平放,手指自然下垂,轻放在键位上,小臂与大臂间略小于 90°。只要桌椅适宜、姿势正确,那么就可以减轻录入中的疲劳,提高录入速度。

(3)掌握击键要领,眼睛不看键盘。打字的关键不在理解而在熟练。如果击键要看键盘,那么就会出现看了稿件看键盘,击键后眼睛又回稿件寻找内容,目光反复移动的情况。这种情况除浪费时间外,还容易造成眼睛疲劳,丢字串行等问题的出现。因此在录入时应做到眼睛不看键盘,手能不加思索地,自动地把所看字符“弹击”出去。这里说的“弹击”键码是击键的要领。录入时两手指尖接触键码的部位不是指甲,而是指端的圆肚部分。看清字符后就应用手指果断地“弹击”相应的键码。弹击键时点到为止,不要用力太大,弹击键码后手指迅速退回原位。击键要均匀有节奏。

(4)掌握基本指法及键位。通常人们把字母键 ASDFJKL 和标点;键称为基本键位(也有称作原位键或导键的)。其中 ASDF 四个键是左手的小指、无名指、中指及食指的原位键;JKL;是右手的食指、中指、无名指和小指的原位键。开始击键时,每个手指应轻放在该指对应的原位键上,每击一健后,手指应迅速返回原位键。每个手指除有它的原位键外,还有其范围键。按键盘码分布,左手小指(A 指)击键范围在 ZAQ1 和左边的其它键,左手无名指(S 指)应打 XSW2,左手中指(D 指)打 CDE3,左手食指(F 指)打 VFR4 和 BGT5;右手食指(J 指)打 MJU7 和 NHY6,右手中指(K 指)打, KI8,右手无名指(L 指)打。 LO9,右手小指(;指)打/;PO 和右边的其它键。空格键根据各自的习惯由两手的大拇指来打。上述是指法的基本内容,只要掌握好,就可提高击键速度和准确性。

4. 数据录入的全过程包括哪些步骤?

数据录入全过程大致可分为六步:

(1)明确录入要求

接受录入任务,首先要弄清录入要求。这里所指录入要求主要包括:估计字符数量,作业成品的准确率、作业交货日期、文件类型与格式、输出作业成品的方式等。只有明确录入要求,才能制订合理的作业流程,把握住作业质量和作业进度。

(2)选用合适的录入程序

对于不同格式的数据应选用不同的录入程序,如

果没有合适的录入格式要编制录入程序。一般情况下，①对于输出的成品是打印纸，原始材料是一般公文资料的录入作业，可借助于中文(或英文)Wordstar等字处理软件来录入或编辑，打开文件的命令采用D。②对于非公文资料的录入作业，如计算机语言的源程序，则应使用Wordstar中的N方式打开文件，或考虑使用行编辑软件EDLIN来录入。③对于由多个字段构成的记录，应采用关系数据库dBASE来完成录入。④对于录入作业中的一些特殊的或专用的格式要求，可在使用上述软件录入后进行单独处理，或自己编制专用录入软件来达到要求。

总之，录入软件的选用，既要考虑录入时，使用方便，提高速度，减少错误，又要考虑后期数据处理和输出成品的格式要求等。

(3) 键盘录入

键盘录入是录入的主要工作。为此录入人员要熟悉键盘、指法熟练、严格按录入要求击键。在键盘录入中应注意以下几点：

①对大批量作业，可拆成多批，要编好顺序号，起好文件名，分批录入后再合并。

②文稿注意保存完好，不要丢失，作好每天的工作任务、文件名、字符数等项记录，便于以后核对。

③做好软盘备份。

④保持稳定愉快的工作情绪，克服环境不利因素的影响，集中精力进行键盘录入。

(4) 校对

校对是数据录入不可缺少的一环。常用的校对方法有键校、目校、“盘校”等。

①键校是指录入员在已经输入的文件基础上再输入一遍。这个过程借助于一种校核软件来实现，一般的专用录入设备都配有这种软件。此校对方式最适用于英文或数字型作业。②目校是指用人工进行阅读检测，这种方法目前在中文录入时使用较多，但校对速度慢，准确率不太高。③“盘校”是指“对盘”校核，就是由两个录入员各打一遍文稿，分别存放在两张软盘上，然后借助软件来进行校核。这种方法的本质上是文件比较。常用的比较方式有两种：一种是按记录比较。两个文件的同一记录有不一致的地方就显示出错误信息，通知录入员检查改正。记录划分可以是一定数目的字符，也可以是一行文稿。另一种比较方式是按字节为单位，把一个文件看成一个数据流，对字符逐个比较。当两个文件中的某一字符不一致时，光标就停在该字符位置上等待修改。以字节为单位进行比较的方法比以记录为单位进行校对速度要快质量要好。一般情况下进行“盘校”后文件大面积错误率可控制在万分之一左右，能满足字符准确率方面的要求。

(5) 数据处理

数据处理是指对作业的记录结构、章节控制符、字体控制符、表格控制符、排版格式等方面的格式查错处理及文件合并等工作，通常可借助排版软件或专用程序来完成。这是作业录入中质量控制的最后一个关口，

所以要仔细认真。

(6) 输出成品

按作业输出要求，将录入好的数据输出。常用的输出成品方式主要有：①存入软盘、磁带等磁介质中供使用；②由打印机打印成文稿供使用。

以上是数据录入工作的主要步骤，在实际中有的步骤要重复或交叉执行。例如，经校对后发现较多错误，有时仍要再进行键盘录入，因此录入人员需注意提高素质和技能，减少错误，降低误码率，尽量避免或减少重复工作。

5. 数据录入中常用哪些字符编码？

数据录入业说的“数据”包括数字、字母、汉字和各种符号，统称字符。对于计算机系统来说不能“直接处理”这些字符，而是处理代表它们的数码，即代码。把这些字符按照某种规律和约定，编成一组数码，就是字符编码。在数据录入中常用的字符编码有中西文之分。西文字符常见字符编码是ASCII码、EBCDIC码；汉字常见字符编码，不同国家有不同代码，我国汉字国家标准字符编码是GB2312—80等。

(1) ASCII码是英文American Standard Code for Information Interchange的缩写，意思是美国标准信息交换代码。标准ASCII代码是7位二进制代码，能表示128种字符，取值范围用十进制数值是0到127，对应十六进制数值是0到7F。ASCII码的编码方式是有序的，0至9十个数字在英文字母前面，52个英文字母按先大写后小写及字典顺序排列；其余还包括32个控制符和34个专用符号。编码和字符之间是一一对应关系。扩展的ASCII码为8位二进制码，十进制数值从0至255，对应十六进制数值0到FF，共256个。IBM PC及其兼容微型计算机均采用ASCII代码。

(2) EBCDIC码是英文Extended Binary-Coded Decimal Interchange Code的缩写，意思是扩充的二进制十进制交换代码。它可以表示英文字母、数字、符号及控制字符等256种字符的代码，EBCDIC码也是有序代码，作用类似ASCII码，主要用于美国IBM公司设计的计算机。

(3) GB2312—80是我国1981年颁布的《信息交换用汉字编码字符集——基本集》的国标号码，它是我国用两个字节代表一个汉字和非汉字图形字符，用以作为计算机系统之间汉字信息交换用的标准代码。该代码表纵向分成94个区，横向将每个区分成94个位置，区号由第一字节标识，位号由第二字节标识。十进制数表示的区号和位号构成了每个汉字或汉字图形字符集的“区位码”。字符集的排列顺序是：1—9区为682个符号；10—15区空，未使用；16—55区为一级汉字，3755个，属常用字，按汉语拼音排列；56—87区为二级汉字，3008个，属非常用字，按部首排列；88—94区空，未使用。国家标准GB2312—80是汉字信息处理的基础代码标准之一，目前已广泛用于我国通用汉字系统的信息交换及硬软件设计工作中。

6. 汉字输入方法是如何分类的?

汉字输入计算机的方法有多种,但可以分为两大类:一是通过计算机键盘人工输入,二是让计算机自动识别。计算机自动识别可以是利用计算机抽取汉字字形特征,实现对汉字的手写体或印刷体的自动输入;也可以是利用语音分析技术,抽取汉语语音特征,实现对汉语语音的自动识别。这种自动识别都由机器将字形或读音转换成代码,然后加以识别,以达到将文字或读音输入计算机的目的,这种方法目前在我国使用还不普遍。

通过计算机键盘,人工键入汉字编码进行输入是目前应用最广泛的汉字输入方法。根据数据录入人员获取汉字编码信息特征又可将汉字输入编码分为音码、形码、音形码三类。①音码是指采用声、韵、调的信息特征,用全拼、双拼、简拼、混拼等方式键入汉字,通过词、词组乃至语法、语义、语用知识区分同音词。例如CCDOS中使用的缩简拼音法、联想汉卡下使用的全拼音及双拼双音法等属于此类音形码。这种码对于熟悉汉语拼音的人是很容易使用的,但重码多,不易盲打。②形码是指采用整字、字根、笔画、笔画组、字形结构等信息特征,利用专用键盘或通用键盘将汉字输入计算机中。五笔字型、长城笔形码等属于这一类。③音形码是指采用音和形两方面的信息给汉字编码。例如声韵部形码、见字识码等属于此类。这些编码方案各有千秋,所以使用时应注意选用较好的输入方案。

7. 字处理软件 Wordstar 中英文版本有何异同点?

英文版 Wordstar 是一个较成熟的字处理软件,适用于各种文书资料的输入、编辑、存贮和打印工作。具有①编辑功能,对输入文本的任何部分可随时进行修改、插入和删除;②查找和替换功能,对正在编辑的全文可快速查找或替换指定的字或短语;③显示功能,能在屏幕上打入文本的任何一部分供编辑修改;④文本块功能,能将单字、句子、段落、整个文件进行块移动、复盖、删除等;⑤屏幕格式编排功能,能设定边界和行宽,将标题居行中位置,重编段落等;⑥打印功能,能进行页面格式设计;⑦菜单提示和求助功能,根据用户需要提供详略不同的菜单及操作说明信息;⑧文件存贮功能,可在操作的不同时刻将打入的文本等以文件形式存入磁盘;⑨邮件合并功能,可由同一封信产生发往不同地址和不同收件人的信件,并能进行文本插入及文件连接;⑩校对拼写功能,能利用字典对全文的单字自动检查拼写错误,作错误标记及改错等。因此,英文版 Wordstar 是在西方国家和我国西文字处理工作中使用最早,应用最广泛的字处理软件之一。

为适应我国汉字信息处理的需要,机电部六所等单位西文 WS 进行了适当改造和汉化工作,产生了 C-Wordstar 汉字文字处理软件,简称 CWS。CWS 的基本功能和操作命令与西文 WS 基本相同,其差异主要有以下几点:①CWS 不具有汉化校对拼写功能;②由

于 CWS(11 行)屏幕上显示 1 个汉字,需占用西文显示时的 4 个字符位置,所以屏幕显示功能不如西文 WS 方便;③由于一个汉字在屏幕上显示宽度占西文显示的两个字符位置,所以 CWS 的标尺行宽度的一半等于该行汉字的个数,并避免半个汉字的出现,标尺行宽度应取偶数;④西文文本中单字由字母组成,单字间由空格或标点隔开。中文文本中字、词间不一定有间隔,所以在使用 CWS 中的 ctrl+A,或 ctrl+F 等以字为单位的移动光标命令时容易产生错误的光标定位,上述命令不易使用;⑤由于汉字打印驱动程序的使用的差别,使得在西文 WSF 控制特殊打印效果的标记的操作命令有的失效,有的有较大变化,因此在打印汉字信息时,要注意参考所使用打印驱动程序的有关说明。

8. 字处理软件 WS 使用中常见术语的含义是什么? 怎样使用?

中英文 Wordstar 有许多特性可以使文本输入工作简单化。为叙述方便,操作手册上采用了一些术语,这些常见术语的含义和简单使用方法如下。

(1)文本文件——在编辑中,需要遵循通常的文字书写规则(如自动换行、分页等)的文字资料文件在 Wordstar 中称为文本文件。常见的有书信、文章、论文、报告等。

(2)非文本文件——在编辑中,不能根据通常格式规则自动处理的数据资料文件在 Wordstar 中称非文本文件。常见的有表格、源程序等。

由于 Wordstar 将通常的文书资料分成文本文件和非文本文件两类,所以在打开文件时方式也不一样,在 WS 开启菜单中使用 D 打开文本文件,使用 N 打开非文本文件。两类文件的存盘命令没有区别。

(3)字绕转——输入文本时,当输入字符超过标尺右界时,WS 自动处理换行的过程称为“字绕转”。CWS 的标尺长度省缺值是 70,即每行可书写满 70 个字符或 35 个汉字,超过时实现“字绕转”。标尺长度可使用 ctrl+O+R 或 ctrl+O+L 改变,改变长度时要注意。输入中文标尺长度为偶数,否则产生字绕转会现半个汉字。

(4)对齐——当输入一段文本时,字绕转在字之间自动插入适当空格,使文本右边成一直列,此过程称为“对齐”。这里值得注意的是:在使用 CWS 时,进行汉字和非全中文方式的 ASCII 字符混合输入,行末的 ASCII 英文单词或数字不实现“对齐”,需要采用人工干涉。

(5)硬空格——按空格键产生的空格称为硬空格。硬空格是文本文件的一个组成部队,只有使用删除命令才能删除。段落开头的空格可由硬空格产生。

(6)软空格——软空格是在字绕转或文本重排时产生的,WS 能有选择地自动插入或删除这些软空格。

(7)硬回车——每段结束时按回车键产生的回车称为硬回车。WS 屏幕右边标志列上有标记“<”。使用

ctrl+B 命令重编文本时,是以硬回车为单位进行的。

(8)软回车——由字绕转自动形成的,WS 屏幕右边标志列为空白。

编辑文本文件时,在输入自然段落结束时打回车键,即使用硬回车;段落内的换行由字绕转产生的软回车实现换行。编辑非文本文件时不能用软回车,由字绕转实现自动换行,必须每完成一行文本输入就打一个回车键。

(9)重排文本——又称重排段落,是指按屏幕标尺将修改排列不整齐的段落进行格式排版。两个硬回车之间为一个段落,排版后每行宽度由按下重排命令 ctrl+B 时的屏幕标尺给定。使用 ctrl+B 前应将光标移到要排段落首字符前,使用 ctrl+B 后光标自动停在下一段开始位置,可连续使用 ctrl+B 重排若干段落。

(10)圆点命令——插在文本中的打印命令。它的标志是用圆点(·)开始于文本行的第 1 列,点后跟两个英文字母,及一个空格或适当参数。圆点命令占一行,由硬回车结束,其功能对每一打印页都起作用。

9. Wordstar 有哪两种重要的编辑功能?

Wordstar 的两种重要的编辑功能是指对文本内容进行查找与替换和文本块操作。

(1)文本内容的查找与替换

在编辑文本过程中进行查找是指从文本中找出指定的字符串,字符串总长度不超过 30 个字符(或 15 个汉字),使用 ctrl+Q+F 命令实现;替换是指从文本中先找出指定字符串,再用给定的内容替换查找到的字符串,使用 ctrl+Q+A 命令实现。查找和替换命令中有六个选择项,可灵活组合使用。

(2)文本块操作

文本块是预先用命令定义的文本内容的一部分。一个文本块可以是一个句子、一段文章或整篇文章。文

本块简称块。块大小是指以块首标记到块尾标记所包含的文本内容。块命令可以简化许多编辑工作。Wordstar 允许使用块命令对文本块进行移动、复制、删除等。

10. 数据录入用哪个软件最好?

该问题最常见。有人说 WS 好,有人说 HW 好,也有人说 PE 好,……,众说不一。

在数据录入中究竟选用哪个软件最好,我们认为不能一概而论,建议从以下几方面去考虑。①所选软件要能满足录入作业的要求,特别要考虑利于作业录入后期的数据处理工作。②所选软件要操作方便灵活,编辑功能强,增加、删除、修改、打印等性能好。③如果录入中文作业所选软件汉化程度要好,适合使用录入人员所选用的汉字输入方案。④如果录入表格方面的作业所选软件制表功能要强,操作要方便。⑤所选软件价格是否便宜,录入人员是否熟悉等。

目前在 IBM PC 及其兼容机上有多种编辑软件、字处理软件、窗口软件等可用来做数据录入,如前面提到的 WS、HW、PE、EDLIN、dBASE 等都可以回来做录入软件,究竟用哪个还要根据录入作业的要求和录入人员使用的录入环境确定。如:长城系列机使用 HW 非常方便,而没有安装长城的微机只能考虑用 WS 等软件;又如联想汉字系统上配备的汉化 PE,编辑功能强,打印文稿方便自如,但没装联想汉卡也就不能使用;又如有些数据库使用 dBASE 作录入软件十分方便。总之,初级程序员应在熟悉两种编辑软件的基础上,尽可能地去多了解和掌握一些其他常见编辑软件、字处理软件、窗口软件、工具软件的使用方法和技巧,这样对复杂的数据处理作业就可以方便地确定使用一个软件或使用几个软件交叉完成数据录入。

个人转让 Apple— II 计算机 1 台

配置:主机,软盘驱动器及卡,彩色显示器(NEC),软件及资料。保证质量。

价格:2200 元

联系人:湖南省长沙市省电子分析所 王大范

邮编:410001

转让 CEC— I 中华学习机

配置:CEC— I 型中华学习机(陕西产) 1 台

软盘驱动器(日本原装) 1 台

Z80 卡 1 只

可编程音响合成器(CEC— PSG) 一只

(有使用说明书)

价格:1750 元(购机时间一年,如有意者赠送全套有关书籍,数张软件)

联系人:陕西省安康市张岭水电三局科研所 李宗卫

邮编:725011

(上接 45 页)

内部的两个小槽中,此时恢复弹簧键体的原状再用。

“3”的方法,用万用表检查弹簧键体是否已修理好。

7. 如果还有其它失灵的键体,都可照以上办法进行修复。全部修好后,照拆卸的相反顺序依次装回主

机,你就能正常使用。

只要发生在弹簧键体内的毛病,如:弹簧弹性变差等,都可用此方法进行修复。对于不能修复的弹簧键体,可用不常用的键体进行对换,如 TAB 键。注意:只换键体,不换键帽。



维修经验谈

软盘驱动器机械故障的分析与维修

广西桂林工商银行计算中心 魏学则

软盘驱动器故障大体上分为机械部件故障和电路器件故障,而机械故障率占软驱故障的 90%,也就是说软盘驱动器故障有 90%是属于机械方面出现异常。

这些故障归纳起来有以下几个方面:

(一)磁头脏

由于长期使用不按时用清洗盘清洗磁头,或使用质量较差的软盘和受潮,发霉的软盘,造成磁头上积磁,污物太多,影响软驱的读写功能。在读写软盘时主机显示器上显示该软驱读错误或写错误,出现这种故障后再去用清洗盘清洗磁头已不能解决问题。应将软驱从主机上取下,用棉签蘸上录音机或录像机清洗磁头用的高级磁头清洗剂,反复轻擦上下磁头,上磁头可用小镜子观察清洗情况,下磁头很直观,擦至光亮无污物即可。注意擦洗过程用力不易过重过猛,以免使磁头变位。

(二)磁头偏

磁头偏一般都是上磁头偏,下磁头偏的情况较少。这种故障现象表现为对一块装有 DOS 内容的盘片,DIR 可以正常显示目录,但执行某一文件时,出现该软驱读错误,按 A、R、I 键绕过错误,这种故障也是因为使用发霉或较脏的软盘,这些软盘表面有污物已失去光滑感,磁头落在上面,盘片高速转动将磁头带偏。

判断软驱读写错误故障是否属于磁头偏这一类的方法是:找一块新软盘插入有故障的软驱,格式化该软盘,之后拷贝一些可执行文件,再去执行某一文件。如正常则说明该写电路没问题,此故障属磁头偏。

借助猫眼盘或磁头定位程序将磁头重新调试定位是最好不过。如果没有这些工具,可凭经验手动定位。我们知道盘片是顺时针旋转,那么磁头多偏向右边,用手指扶到软驱上磁头,轻轻往左带,边调边试,直至正常读写标准盘。

如果上述方法不能解决,就放松磁头小车上固定磁头支架的两颗螺丝,将磁头向左推 0.5mm 左右,然后试读写盘片,要细心调试,动作不可太大。

(三)上磁头不到位

当软驱插入盘片压下仓门杠杆后,磁头落不到盘片上,出现这种故障其现象与磁头偏相类似,只能列出目录而不能读,写其中任意文件,或格式化时只有 0 磁头工作,显示“0”头某某道,而不显示“1”头某某道。出现这种故障时,把软驱拆下,不插盘片关上磁盘仓门,可以以软驱侧面观察到上磁头在下落到一定位置后就不再下落,与下磁头保持有 5—7mm 的距离,这样上磁头是不可能读到信息的。

我们从软驱上面观察,当你将软盘仓门杠杆压下后,联动上磁头起落杆也落下,使上下磁头相吻合。出故障时上磁头起落杆下落不到正常位置,致使上磁头落不到位。

处理方法是:在软盘仓门杠杆压下的情况下,用手去压磁头起落杆至上磁头刚好落到下磁头上。然后在杠杆联动的各轴上加注少量缝纫机油。正常情况下,上下磁头与盘体之间的压力是靠磁头与支架之间的弹性簧体来调节的。不必耽心磁头对盘片的用力过大或过小。但两磁头在读写时要相吻合。

(四)读、写时,运行到内道出错:

软驱在读写时,1.2MB 软盘走到 70 道左右,360KB 软盘走到 30 道左右开始出错或磁头小车出现“咔”响声。

这种故障出在磁头小车运行轨道上。在软驱被面可以看到有两条光亮的园轴,磁头支架在其上面前后运动。这轴上积有灰尘后,被磁头小车推到轨道的顶端,久而久之轨道顶端的灰尘将磁头小车卡住在某一部位,使其不能运行到轨道的顶端。造成读错误,或写不满盘等。

清理两轴上的灰尘,加注少量缝纫机油,故障即可排除。

软盘驱动器,无论任何型号,其电路部分各有所异,但机械部分都相类似,所以大家在检查一台有故障的软盘驱动器时,首先判断是电路故障还是机械故障,然后把故障压缩到某一部位,修起来也就方便了。

更换集成块一法

吴广元

在维修过程中,更换印刷电路板上的集成块是一件很麻烦的事。下面介绍的方法,无需专用设备,就能

很容易地更换集成块,且不会损坏电路板。

一、在离电路板尽可能高的位置剪断待换下集成块的全部管脚。

二、用烙铁逐个溶化电路板上已剪断管脚的焊锡,并用尖嘴镊子将管脚拔出。

三、准备若干根不沾锡且熔点高的小细棍(木、竹等),如果手头暂时没有,用大头针代替也可以。用烙铁

依次溶化电路板焊孔中的锡,将准备好的小细棍插入电路板上的焊孔中。

四、锡凉后,拔去电路板焊孔中的小细棍,焊孔就

完全清理干净了。

五、插入待换上的集成块,用烙铁焊好并剪齐管脚即可。

APPLE- II 机内存故障 维修方法

滕 达

通常 APPLE- II 机损坏的最大可能情况有两种,一种是主机电源损坏,一种是内存集成块损坏。

内存集成块损坏是指动态 RAM(4116)的损坏。损坏轻者造成图形显示不完整,DOS 调不进内存;重者一开机就启动不了。下面介绍一种方法能准确地查出损坏的集成块(4116),使计算机恢复正常。

APPLE- II 机的 RAM 分为三排:

第一排 C3-C10 地址: \$ 0000 ~ \$ 3FFF

第二排 D3-D10 地址: \$ 4000 ~ \$ 7FFF

第三排 E3-E10 地址: \$ 8000 ~ \$ BFFF

C3-C10, D3-D10, E3-E10, 是主机板上集成块的坐

标。

具体方法如下:

①打开计算机电源,如果不能启动,屏幕上出现杂乱图形,那么应先检查第一排(C3-C10),最简单的方法是替换法,即将 C3-C10 八块 4116 用好的 4116 替换,直到故障排除为止,没有其它简便方法。如果仍不能排除故障,则应检查第六支 ROM(F3)

②如果打开计算机电源后能够启动,但是在作图状态下,屏幕不能完全清屏,那么就说明第二排 RAM 有损坏,地址(\$ 4000 ~ \$ 7FFF)

在 APPLESOFT 状态下输入 HGR2,将屏幕消除,然后 CALL-151 进入监控状态。在监控状态下查看 \$ 4000 的值,如果第二排 RAM 损坏,则 \$ 4000 的值就是非 0,将 \$ 4000 的内容由 16 进制转化为二进制,例如 \$ 4000 的值为 \$ 15 转化为二进制后则为 00010101。这个二进制的八位数分别用第二排的八块 4116 RAM 来贮存。具体的对应如下:

母板坐标: D3 D4 D5 D6 D7 D8 D9 D10

内存单元: 第 0 位 第 1 位 第 2 位 第 3 位 第 4 位
第 5 位 第 6 位 第 7 位

以 \$ 15 为例: 1 0 1 0 1 0 0 0

出现 1 的位置所对应的集成块就是损坏的 RAM,如上例可知。D3 D5 D7 这三块 4116 损坏,只要将其拆下换上了新的集成块即可,其余的类推。

③如果计算机启动后,DOS 调不进内存,使得驱动器不能使用,则有可能是第三排的 RAM 损坏,地址 \$ 8000 ~ \$ BFFF

在 APPLESOFT 状态下输入 HGR 和 HGR2,然后用 CALL-151 进入监控状态。在监控状态下输入,

\$ B000 < 2000, 2FFFM \$ 8000 < 2000, 4FFFM, 将 \$ 8000 ~ \$ BFFF 清为 0,然后读出 \$ 8000 的值,如果非 0,则说明第三排 RAM(4116)损坏,具体判断损坏集成块位置和②相同。

巧修 CEC- I 受损键盘

黄文贵

中华学习机 CEC- I 型键盘上的 69 个键都是弹簧键体。在使用过程中,人们频繁地按动部分键造成个别键失灵,或者发生意外冲击。现象是:按动这些失灵的键时,屏幕上无任何显示。这些失灵的键,完全可以自己修复。现将修理步骤介绍如下:

1. 接通学习机电源,同时按下 CTRL-RE SET-TEST 三个键,最后释放 TEST 键,对主机进行自检,自检内容在《用户使用手册》、《中华学习机 CEC- I 技术参考手册(硬件)》书有所介绍。当自检全部正常时,可进一步确定故障发生在键盘上。

2. 断开学习机电源,旋下外壳上的六颗螺钉,取下外壳上盖,拿出键盘。键盘右侧有一根 26 线的扁平电缆,此电缆与主机相连。小心取下接线卡,使键盘脱离主机,以利于修理。

3. 在键盘的印刷电路板上,每个键的名称都标得非常清楚。用万用表的欧姆档进行检测,两表笔放在失灵键的两个焊点上,注意接触应良好。正常的键,没按动时电阻值 R 应为无穷大(即万用表指针不动),按动时电阻值应为零值。失灵的键,电阻值始终为无穷大,用此方法进一步确定故障发生在弹簧键体上。

4. 取下失灵键上的键帽,即标有 A、B、C、……字母,向外稍用力拉就能取下。用电烙铁、医用的吹耳气球(注:搞摄影的人一般都具有此工具)。去掉弹簧键体两脚上的焊锡。用电烙铁加温,气球吹去焊锡。注意:温度应合适,时间不宜太长,以防损坏印刷板。焊锡一定要去干净,使弹簧键体两脚脱离印刷板。然后,又在键盘表面上用两把中等型号的扁平螺丝改刀放在弹簧键体的两旁,适当用力向上挑,就能取下弹簧键体。

5. 此弹簧键体是密封的,仔细观察形状,你就能很容易拆下键体的外壳。一定要小心拆卸,里面装有一根弹簧和两块铜片,还要注意内部的形状和安装位置,以利于顺利安装。即使搞错,你只需稍加观察,也能装上。

6. 拆开弹簧键体后,你会见到两块铜片,一片像“Π”型,一片像“山”型。据自己的修理经验,认为拆动“山”型铜片较合适,把“山”型铜片的两边向外拨动,目的是使两铜片能良好接触。然后分别把两铜片装入到

(转至 43 页下)



VIRUSCAN 病毒处理软件包功能简介

新书与软件

朱 禹

VIRUSCAN V61 版病毒处理软件包功能强大,它可免疫、检测、清除到目前为止已发现的 82 类病毒及它们的许多变种病毒,是目前处理计算机病毒最有效的软件。

该软件包是由美国 McAfee Associates 发行的,它包括十六个文件(8 个执行文件),它们功能如下:

1. SCANRES——计算机病毒免疫程序,它可以被放置在批处理文件(AUTOEXEC·BAT)中,并驻留内存,开机时它将以 BOOT 扇区、硬盘分区表、隐含文件及各子目录下的所有文件包括其自身进行检测,看是否带有病毒。在计算机正常工作中,每当你装入程序时它对被装入的程序也进行同样的检查,若检查到有病毒存在,它将显示警告信息和所感染病毒的名字。若在你的批处理文件中装入了 SCANRES 程序,你的计算机将免受病毒感染,也就是说你的计算机对病毒具有免疫能力。

2. SCAN——计算机病毒检测程序。它可以检测到已发现的所有病毒。检查的范围可以包括硬盘分区表、BOOT 扇区、文件分配表、子目录下的所有文件及隐含文件。在检查时有多个选择项可供选择。比如:只对内存进行检测,只对 COM 文件进行检测或只对某一病毒进行检测等等。这些选择有助于提高你检测的效率。

3. CLEAN—UP 病毒清除程序。该程序可以清除 SCAN 发现的文件中的所有病毒。在清除过程中有多个选择项可供选择。这些选择可以用一个命令清除多块磁盘上的病毒,也可以清除一个具体程序上的病毒,除此外还介绍了自 57 版以来所发现的各种病毒的症状及主要危害。

4. MDISK——BOOT 扇区、分区表上的病毒清除程序,有些病毒感染了 BOOT 扇区、分区表是不能用 CLEAN 来清除的,对于不同的操作系统要使用不同的 MDISK。它可以是 MD40、MD33、MD32 或 MD30 之一,它们分别适用于 DOS4. 0、DOS3. 3、DOS3. 2、DOS3. 1 (或 3. 0)。

5. NETSCAN 是 VIRUSCAN 的网络检测程序,它可以检测 PC 网络或工作站上的病毒。NETSCAN 有多种选择项可供选择,它可以检测多个驱动器上的多种病毒,也可以对一个驱动器或一个文件进行单独检测,检测出病毒后可以用手工单独的清除,也可以自动地清除多个病毒。

6. VALIDATE——证实程序,它是用于证实上述五个程序的完整性的。VALIDATE 可以检查上述五个程序是否被篡改过,该程序以两种不同的方法产生循环检查以确保其高度安全性,产生的结果可与 McAfee Associates 所公布证实数据进行比较,以确定上述五个程序的完整性。

7. VIRLIST——病毒特性表,在该表中给出了已发现的 PC 机所有病毒的名字,它们传染的范围,被感染文件增加的长度、清除它们的方法和它们的破坏性。

8. REGISTER——注册说明程序。这一程序中给出了使用上述软件的注册方法,从该程序中可以看出美国软件的使用注册的方法和有关价格,对我们了解美国软件市场的情况有一定的帮助。

VIRUSCAN 的每个执行程序都有说明与使用方法,想进一步了解的读者可与电子工业出版社软件部联系(北京 173 信箱软件部,邮编:100036)。

Microsoft Windows 3.0 窗口软件简介

Windows 3.0 是美国 Microsoft 公司 90 年在原 2.0 基础上推出的窗口软件,它扩充了 DOS 的操作方式,将改变 PC 机用户使用个人计算机的方式,解脱用户记忆理解 DOS 命令以及各种选项的负担。Windows 用下拉菜单和对话框代替了 DOS 的提示符,将各种实用程序表示为图标画面,许多命令可以通过鼠标的移动很直观的完成。尤其对计算机的初学者,提供了非常友好的界面。

Windows 为 PC 机操作系统提供了一个很好的多

任务环境。它充分利用 CPU 空闲时间,允许同时运行多年程序,并为每个程序各开辟一个窗口,且各窗口之间可以方便地交换信息(图象、图形和文字),通过这种信息交换的方式,可以将不同应用程序中所做的作业合并到一个单独的文件里。

Windows 非常适用于办公事务处理,它提供了许多办公经常要做的功能,如资料卡、计算器、日程表及模拟时钟等,完成了办公的自动化。所以,也有人称 Windows 是办公桌系统。

Windows 程序包提供了三个基本软件:write(文书处理)、paintbrush(彩色绘图)和 terminal(终端)。这三个软件较充分地显示了 Windows 的特点。Write 是一个具有图形功能的字处理系统,它提供了多种字体、多种字形,可将图形、文字同时在屏幕上显示,并且与打印结果完全一致,即“所见即所得”。Paintbrush 是一个多功能彩色画面设计软件。使用 paintbrush 可以绘制出具有多种线型、多种背景、多种色彩并可与字符同时显示的画面。这些画面可以在屏幕上被任意移动、翻转、放大和缩小。Terminal 提供了终端仿真、文件的传递与接收功能等。这种功能能使用户的微机可以成为其它计算机的终端,并可以灵活地与其它计算机进行信息交换。

Windows 有着优越的内存管理功能,并且在其中的程序设计与外部设备完全独立的。Windows 不象 DOS 系统,只有经过专门编程才可以使用 640K

以外的扩展内存,在 Windows 环境中,用户可以直接使用扩展内存,最大可以使用到 16MB。在 Windows 环境下编程,可以直接使用该环境提供的各种图形元素(如出口、下拉菜单、控制、对话框、图标、字形变换、字体变换和 Bitmap 等),用在帮助信息生成程序及其它工具,可编写出具有在线帮助的交互式图形界面程序。用 Windows 写出的程序,具有对外设(如显示器、打印机等)的独立性。即同一个程序,可以不加修改地直接用于 EGA、VGA 或 CGA 等显示适配卡上,也可以不加修改地直接使用不同的打印机、硬盘等,而一般的 DOS 程序,总要注明适用于何种显示适配卡、打印机、硬盘等。

总之,Windows 不仅可以使您充分利用计算机的潜力,赋予更多的新功能、新特性,而且其丰富的图形界面将在 PC 机上为您提供更直观、更高效的工作环境。

最新版本 PC TOOLS V6.0

美国第十三大大软件公司(Central Point Software)最新推出 PC TOOLS V6.0 版本,这是一个功能强大,界面友好,使用方便的工具软件,是微机管理与办公自动化必备之工具!

PC TOOLS V6.0 由三大部份功能组成:

- 1)数据管理与维护
- 2)硬盘备份
- 3)桌面管理系统

数据管理与维护包括下列程序模块:

PC SHELL:这是一个在简便易用的窗口环境下,具有全部 DOS 维护命令功能的实用程序,提供拷贝,删除和比较文件功能,文件和磁盘编辑功能,以及目录管理功能,系统盘和数据盘格式化功能。PC SHELL 可按原格式显示大多数数据库建立的文件,制表软件和字处理软件建立的文件。另外,PC SHELL 还可恢复已删除的文件,显示常驻内存分布图,以及在 PC SHELL 状态下运行其它可实用程序,PC SHELL 可以常驻内存,或做为一个单独的实用程序使用。

DiskFix:这是一个可以修复软/硬盘各种问题的实用程序,其功能强大,易于使用,修复工作全部自动完成。

Mirror/Rebuild:用于防止意外删除数据和硬盘格式化。Mirror 在一个专用的隐藏文件中保存文件分配表和硬盘根目录,一旦意外地格式化了硬盘,用户可以快速恢复数据。Rebuild 可以尽可能地恢复 Mirror 所记录的硬盘状态,以便有效地挽回数据删除和格式化磁盘所造成的损失。PC SHELL 的 Undelete 功能可以利用 Mirror 建立的删除跟踪信息,自动恢复被意外删除的文件。

PC-Cache:通过把最常用的信息存在计算机中来提高硬盘的存取速度。

Compress:一个独立的实用程序,用于检查并消除硬盘的存储碎片,它也可以检查硬盘坏区,并把文件移出这些坏区。

PC Format:可格式化硬盘和各种密度的软盘并可利用 Rebuild 恢复这些盘中的数据。

PC Secure:用于保密机密数据和程序。PC Secure 可以在磁盘上加密,解密,压缩和隐藏文件。PC Secure 使用 DES 加密系统,使数据完全随机化,没有密钥实际上是不可能解密的。若使用压缩选项,PC Secure 可以使绝大部分文件所占磁盘空间减少 25%到 60%。

硬盘备份功能由 PCBACKUP 模块组成,可对各种磁盘或磁带进行备份,支持对数据的压缩及自动进行备份的功能。

桌面管理系统具有强大的文字处理及办公自动化的功能,可同时支持 15 个窗口的操作,其中的 Notepads 字处理器提供了类似 Wordstar 的字处理功能,包括文本的剪裁,粘贴和拷贝,可进行拼写检查。Outlines 功能提供了对信息进行结构化管理的手段。数据库功能提供了数据库管理与查询的手段,实现了与 Dbase 和 Foxbase 的数据共享。

活动计划功能对于有效地安排工作日程计划,准时赴约可以有很大帮助。

桌面系统还支持远程通信功能,可使用 Modem 实现文件传送,也可使用 FAX 接口板进行远程传真服务。同时具有 HP-11 科学运算型,HP-12C 函数型,HP-16C 程序设计员和财会型计算器功能。

中国计算机软件与技术服务总公司(中软总公司)CS&S

市场开发部向您提供下列产品:

电脑大厦

长城计算机

产品名称	内存	软盘驱动器	硬盘	显示卡	显示器	零售件
GW0520EM	512K	360K * 1				5950 元
GW0520DH/10	640K	360K * 2	20M	CMGA		9900 元
GW0520/10	640K	360K * 2	20M	CMGA	GW500	12000 元
GW286BH/12	1M	1.44M + 1.2M	40M	CVGA	GW500	16500 元
GW286EX/16	1M	1.44M + 1.2M	40M	CVGA	GW500	22000 元
GW386SX/16	2M	1.44M + 1.2M	40M	CVGA	GW500	26000 元
GW386/20	2M	360K * 1.2M	40M	CEGA	GW500	30000 元
GW386/25C	2M	1.44M + 1.2M	40M	CVGA	GW500	33000 元

艺高(REGO)系列微机

型号	软盘 (KB)	硬盘 (MB)	显示器	显示控制	串并口	主板 RAM 基本配置	零售参考价 (人民币)
0520/10DM	360×2	NO	14"单显 720×350	主板 Hercules	主板 1 串 1 并	640KB	5600 元
0520/10HM	360×2	20MB	14"单显 720×350	主板 Hercules	主板 1 串 1 并	640KB	7600 元
0520/10CE	360×2	20MB	14"单显 1024×768	CEGA	主板 1 串 1 并	640KB	9900 元
0520/10DE	360×2	20MB	14"彩显 VGA 1024×768	CEGA	主板 1 串 1 并	640KB	12000 元
286/12—BH	360KB, 1.2MB * 1.44MB	40MB	14"彩显 VGA 1024×768	CEGA * CVGA	主板 2 串 1 并	1MB	17500 元
286/16CV	360KB, 1.2MB * 1.44MB	40MB	14"彩显 VGA 1024×768	CEGA * CEGA	主板 2 串 1 并	1MB	22000 元
386SX/16	1.44MB 1.2MB 外挂	40MB * 100MB	14"彩显 VGA 1024×768	CEGA * CVGA	主板 2 串 1 并	1MB	24000 元
386SX/20 DV	1.44MB 1.2MB 外挂	40MB * 100MB	14"彩显 VGA 1024×768	SVGA	主板 2 串 1 并	2MB	25000 元
386SX/20 CV	1.44MB 1.2MB 外挂	40MB * 100MB	14"彩显 VGA 1024×768	CVGA * CEGA	主板 2 串 1 并	2MB	26000 元

打印机 LQ1600K, 5200 元/台。CR3240, 5800 元/台, AR3240, 6500 元/台。

新教材《SQL 语言初学与入门》配有 SQL 语言中文自学软件 3 片盘, 90 元/套

地址: 北京海淀区学院南路 55 号电脑大厦, 邮编: 100081, 乘 16 路汽车电脑大厦下车 电话: 831.7722—1319

开户行: 中国工商银行海淀分理处, 帐号: 461187—79 联系人: 温友良 寻呼: 126 呼 3228