



電子

ISSN 1000-1077

與 電腦



• ELECTRONICS AND COMPUTERS •

欢迎订阅 《电子与电脑》



● 国内代号：2-888 定价：0.95元

OKI 公司单片机资料

低功耗

高速

OLMS-65 系列

型号	ROM (字节)	RAM (半字节)	电源消耗	LCD 段数
MSM6502	2000	128	100 μ A	108

OLMS-64 系列

型号	ROM (字节)	RAM (半字节)	机器周期	I/O 口
MSM6402	2048 \times 8	128	1 μ s (4MHz)	36
MSM6404	4096 \times 8	256	1 μ s (4MHz)	36
MSM6411	1024 \times 8	32	1 μ s (4MHz)	4
MSM6422	2048 \times 8	64	3 μ A (4MHz)	18

OLMS-50/60 系列

型号	ROM (字节)	RAM (半字节)	电源消耗	备注
MSM5052	1280 \times 14	62	3 μ A	在片 A/D 连接器 52 段 LCD 驱动器
MSM5052	1020 \times 14	62	3 μ A	88 段 LCD 驱动器
MSM5052	1792 \times 14	96	3 μ A	120 段 LCD 驱动器
MSM5052	1792 \times 14	90	3 μ A	可接太阳能电池 76 段 LCD 驱动器
MSM5052	2560 \times 14	120	3 μ A	189 段 LCD 驱动器
MSM5052	2048 \times 8	64	6mA	内附 DTMF

OLMS-40 系列

型号	ROM (字节)	RAM (半字节)	机器周期	备注
MSM5840H	2048 \times 8	128	8 μ s (4MHz)	I/O 口; 40
MSM5842	768 \times 2	32	8 μ s (4MHz)	I/O 口; 21
MSM58421	1536 \times 8	46	8 μ s (4MHz)	35 段 LCD 驱动器
MSM5847A	1536 \times 8	90	600 μ s (32KHz)	72 段 LCD 驱动器

公元 1993 年

一 JANUARY 月

S	M	T	W	T	F	S
					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						

二 FEBRUARY 月

S	M	T	W	T	F	S
	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						

三 MARCH 月

S	M	T	W	T	F	S
	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			

四 APRIL 月

S	M	T	W	T	F	S
				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	

五 MAY 月

S	M	T	W	T	F	S
						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					

六 JUNE 月

S	M	T	W	T	F	S
		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			

七 JULY 月

S	M	T	W	T	F	S
				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

八 AUGUST 月

S	M	T	W	T	F	S
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				

九 SEPTEMBER 月

S	M	T	W	T	F	S
			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		

十 OCTOBER 月

S	M	T	W	T	F	S
					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						

十一 NOVEMBER 月

S	M	T	W	T	F	S
	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				

十二 DECEMBER 月

S	M	T	W	T	F	S
			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	



電子

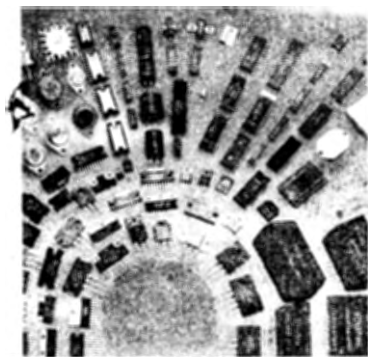
ISSN 1000-1077

與電腦

12

一九九二年

总期第93期



• ELECTRONICS AND COMPUTERS •

電子與電腦

目 录

• 综述 •

- 国外消费电子业 吕问黎(2)
应用计算机网络辅助高中物理教学
..... 肖美 王天谔 李燕萍(3)

• PC 用户 •

- 也谈 FoxBASE+弹出式菜单的实现方法
..... 王瑞华(5)
C 语言如何读取 dBASE 的库文件 李晓华(6)
如何实现题库系统的图文并貌 裴伟东(7)
一个简单实用的磁盘加密程序 任绥海(9)
2.13HZ 汉字系统安装虚拟盘字库新探 廖 凯(10)
怎样从死循环程序返回 DOS 方 晨(10)
BASIC 实现“卡拉—OK” 于龙滋(10)
实现 1.2MB 软磁盘之间直接拷贝简法 何崇乐(12)
兼容机怎样运行 BASICA.COM 邓文超(13)
香港病毒的消除和防范 孟 桥(14)
谈数据文件“死而复生”的技巧 李瑛彬(15)
数据自动存盘程序 汪海波(16)
PC 机软件加解密技术剖析(续) 李文亮(18)

• 学习机之友 •

- 推荐一个扩展 BASIC 语句——拆字串语句
..... 王 凯(20)
也谈内存信息的打印 张 亭(22)
增强中华学习机的音响功能 杨云霄(23)

计算各类债券储蓄收益利息 BASIC 程序

- 陶文庆(24)
哥德巴赫猜想 BASIC 程序的改进 卢良红(24)
数组的动态删除 冯端品(25)

• FORTH 语言讲座 •

- 第一讲 概述 丁志伟(26)

• 学用单片机 •

- 模/数、数/模转换实验(续) 张俊谟(29)

• 电脑巧开发 •

- 中华学习机巧测电容 邓本富(33)
家庭如何配置中华学习机 聂铁轮(34)

• 电脑游戏机 •

- 第四讲 游戏程序的设计过程(中) 于 春(35)

• 维修经验谈 •

- LQ—1600K 打印机接口专用集成电路 M54610P
的简易修复方法 赵继文(38)
GW286 计算机硬盘控制卡故障处理一例
..... 杨远成(40)
TP801 单板机的常见故障 花 成(40)

• 读者联谊 •

- 普及型 PC 个人用户软件交流联谊活动问题解答(十二)
..... 王路敬(41)
培训消息 (43)
1992 年总目录 (45)

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:北京报刊发行局

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:张 丽

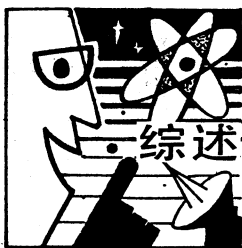
订购处:全国各地邮电局

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



国外消费电子业

吕问黎

1991年的美国消费电子业领域中有多项技术上或商业上的成功,其中视频压缩技术和位率压缩技术是最引人注目的两项。它们已经成为美国标准清晰度电视传输系统的最佳的备用技术。有线广播的管理者们也已做好准备把这两项技术运用到传统的NTSC制式系统中去,以便在现有的电缆条件下增加新的电视频道。

数字化音响技术也得到了广泛的开发。这项技术可以消除微弱的杂音而保留较明显的声音信号。飞利浦公司就使用它的新产品“数字式压缩盒”系统去掉了人耳听不到的声音信号,以减少高保真声音存储的位率。索尼公司在它的“微型唱片”系统中也这样做了。这项技术还将被运用到数字化广播和个人电脑电子通信领域中的多媒体应用产品方面。

高清晰度电视

几种先进的电视传输系统在技术上的进展终于在1991年7月弗吉尼亚州亚历山德拉的“先进电视测试中心”展示在人们面前。第一个被测试的“先进兼容电视系统”(ACTV)是由戴维·斯纳夫研究中心、NBC电视网、飞利浦电器公司和汤姆逊电器公司联合发起和研制的。这几家公司成立了一个支持“先进兼容电视系统”的国际财团:先进电视研究财团。这个财团还同时正在发展数字化的无线电与电视联播的高清晰度电视系统,这个系统也正在准备参加测试。其他已经测试过的系统还有Digi Cipher系统和日本广播公司的Narrow Muse。前者是通用仪器公司、马萨诸塞技术研究所等机构组成的美国电视联盟(HTVA)研制的。按照计划,今年还要参加测试的有Zenith电子公司和AT&T微电子公司联合开发的DSC高清晰度电视系统以及麻省理工学院与美国电视联盟生产的“ATVA高级系统”。

这些参加测试的系统必须能够完整地传播电视音响和画面,操作必须是实时的。必备的硬件包括电视台方面对电视信号进行编码和调制的全部部件,以及电视机方面的解码器和解调器。

每个系统在实验室内的测试时间约为7周,包括对共有频道测试(非常邻近的地区使用同一频道却装载不同的节目)、相邻频道测试(两个接近的频道装载不同的节目)以及对在飞机上电视图像的颤动的测试等。实验室测试主要集中在每个系统在传输信号过程中的缺点方面,如各种类型的电子噪声、环境噪声、由于信号传输路线的地面状况和建筑物的影响而产生的

多路传输中的重像现象等。系统还要在不同的界面和干扰条件下录制数字化的录像带,然后由非专业人士来放映,以做出主观的测试。

实验室测试完毕以后,联邦通信委员会的顾问委员会计划对占优势的系统进行实地测试。把这两项测试结果加起来,再考虑到法律、经济、公共政策等方面的因素,联邦通信委员会将在1993年提出一个新的作为标准的系统。

在欧洲和日本,高清晰度电视也正在蓬勃发展之中。欧洲委员会去年颁布了一个适用于卫星广播的指导性文件,鼓励使用D₂-M₆₀标准(这是一个提高分辨率的过渡性的标准),作为走向高清晰度电视的一步。但是,据1991年9月的《经济时代》报道,虽然法国和荷兰政府支持这个系统,法国的汤姆逊、荷兰的飞利浦和芬兰的Nokia三大公司却持反对态度。

91年10月起,日本开始每天播放8小时的超过一般分辨率的电视节目;宽屏幕的NTSC制式电视机则作为日本提高分辨率努力的另一部分投放了市场。这些过滤性措施之后,也必然是数字化的高清晰度电视。

提高电缆容量

在美国,提高有线电视电缆的频道容量的倾向正在发展。例如在纽约昆斯区的白石(Whitestone),“时代华纳有线电视”已经把550兆赫75频道的有线电视网换成了一个1000兆赫150频道的系统;在纽约的Flushing地区的总台与白石分台间安装了一条光导纤维;还把以往在电缆网络的每一条支线上都有的放大器全部取消,改为在每个街区至多装4个1000兆赫放大器。由于频道增多,电视网节目更加丰富;由于放大器减少和光导纤维的使用,信号传输可靠性增加了,噪音和失真减少了。

这些改进只是提高电缆功能的开始,通过压缩技术还可以得到更多的频道。例如康涅狄格州美国电视与通信公司副总裁沃特在去年6月伊利诺斯州的一次国际会议上提出了一种方法,按照他的方法,可以将普通的1000兆赫系统的150个频道分为三部分:一部分为60个,传输普通的NTSC节目;另一部分30个,传输高清晰度电视节目;剩下的60个用5:1的压缩法产生300个频道,从而使这个1000兆赫系统共有390个频道。

去年秋天,有线电视实验室(Cable Labs)、远程通信公司、Viacom电视网和弗吉尼亚州亚历山德拉的公

共广播网联合发出一项声明,希望能够得到传送数字压缩节目的网络设备。数字压缩系统之所以有如此吸引力,还因为在今天的太空时代,如果有了优秀的压缩技术,只需一颗卫星就可以将众多节目传向四面八方。

数字式声音存储

虽然真正意义上的数字式音频磁带还没有上市,但已经有两种相互竞争的系统出现了。一种是飞利浦的“数字式压缩盒”;另一种是索尼的“微型唱片系统”。

飞利浦的系统可以录制新式的数字式磁带,播放这种磁带或传统的磁带。新式磁带与普通磁带大小相同,当它以最高速度运转时,录制声音信号时间为90分钟。

系统的编码技术称为“精密自适应分波段编码”,是整个产品的核心。它录制声音信号的密度可以比普通脉冲信号调制法大4倍。它的动态范围是108分贝(16位),杂音低于0.0025%,频率范围从5赫兹到22千赫。在录制时,声音频率范围被数字式地分为32个分波段,信号处理器首先根据不同的频率确定声音的最低录制音量起点(也就是只有这个音量以上的音才录,其余的全部舍弃。对人耳来说,对3.4千赫这种中等频率的声音最为敏感,所以处理器将中等频率的起点音量定得最低),然后处理器在录制时动态地适应着不同的音量起点。正由于大量低于起点的无用信号被丢弃,编码速度很快,磁带的容量也很大。这样编码得到的信息,伴随着纠错码,被多路传输到16个185微米宽的磁道上,并列的16个数据磁道加上旁边的辅助磁道都容纳在3.88毫米宽的磁带上。

索尼的“微型唱片系统”可以在64mm的光磁唱片上录制74分钟数字式音频信号,这比标准的音频压缩唱片要多一倍。它使用的压缩技术称为“自适应变换音频编码技术”。

这个系统是在44.1千赫频率用16位量化,对声音进行数字化。编码器首先将一段20毫秒的声音分成1000个小区,然后把每一个小区(约0.02毫秒)用16位数形式来表示其声音的特征。接着,编码器将人耳最低听力限度、声音互相遮蔽效果等针对人耳能否听到的因素考虑进去,略去没有用的数据,最终达到压缩成只有原始信号的1/5。

此外,索尼这个系统在播放时还有一个特殊设计,有一个可以存入三秒钟声音的1兆字节小存储器,当发生剧烈震动一类短暂的不能读取声音信号的情况时,系统仍可以正常放音三秒钟。

收音机广播的压缩技术

数字压缩技术成为一大潮流后,不仅对有线电视、录音系统产生了影响,而且也波及收音机广播,虽然这方面的发展远落后于前两者。91年10月,在美国电子与电子工程师协会的纽约会议上,新译西州洛克维尔的CCS公司总裁阿尔蒙·克莱格预测了数字式收音机的前景,他说:很多广播公司已经准备向听众提供无

噪音、无干扰的数字式高质量节目,既然采用了数字式,当然压缩技术就大有用武之地。他还认为,这种广播的频率分配可能有三种方式:一种是在一定范围内(比如1.5千兆赫)实行重新分配;一种是在现有AM和FM波段内分配频率,但必须考虑到数字式广播肯定比原有的广播波道要宽;最后一种是在FM的200千赫内,普通广播与数字广播混同使用。当然,这一切都有赖于联邦通信委员会做出新的决定。

应用计算机网络辅助 高中物理教学

北京和平街一中(100013)

肖 美 王天谟 李燕萍

高中物理尤其是高一物理比较抽象,是历届学生深感难度极大的一门主课。多年的实践证明,采用计算机进行辅助教学,把抽象的、不易观察甚至无法观察的物理现象进行形象逼真的动态模拟,已经在提高学生兴趣、培养学生分析问题和解决问题的能力方面都取得了较好的效果。但是,使用分立的计算机进行辅助教学,并不能满足进一步教学改革的需要。这是因为采用分立的计算机进行辅助教学,教师难以控制各台学生机的进程,师生间的信息难以交流,教师难以掌握每个学生使用计算机学习的实际情况,从而难以进行管理和组织教学。为了进一步为计算机辅助教学创造更好的环境,我们开发了计算机教学网络,将教师机和多个学生机联网,实现资源共享和信息交流,保留并发展了传统教学和分立计算机辅助教学的优点,基本形成了一种新的辅助教学模式,经过多次实践,取得了预期的效果。

一、应用计算机网络辅助教学的过程

我们从今年开始,在高一物理课上,用计算机教学网络这个辅助教学手段,参与了“力的合成”、“运动的合成”、“物体的运动”一章复习课,“振动和波”等等物理教学课程,现以高一物理必修本第一册第八章第2节“气体的等温变化、玻—马定律”为例谈谈计算机网络在物理教学中的应用。

课程在计算机房进行,每两个学生一台计算机,共24台学生机,一台教师机,讲桌上有投影仪及玻—马

定律演示器。

教学过程大约分为六步：

(一)教师提出若气体的状态发生变化时,应如何研究状态参量的变化规律。人们研究事物的规律是从简单到复杂,因此我们先研究一个状态参量不变(即温度不变)的条件下,气体的另两个状态参量的变化遵循什么规律。

(二)先用玻一马定律演示器粗略地、定性地给学生演示实验,由于仪器较少,学生离得远,因此学生根本看不见仪器上的读数,只能大致看看水银面上下移动情况,粗略知道压强体积是增大还是减少,但具体改变多少看不见。

(三)由于上述仪器的演示看不清,因此用计算机细致地模拟整个实验的动态过程。由教师在教师机上操作,通过网络发送到每台学生机,教师一边操作一边讲解,学生通过屏幕看到了形象清晰逼真的整个实验过程,读出了各个状态的实验数据,比较了每次实验结果。

(四)教师由实验结果得出玻一马定律,总结出公式,讲解使用条件。由于每个学生都由计算机详细地观察了实验,这样,由实验上升为理论,对玻一马定律的理解就较容易。

(五)教师向学生发送基本训练题,每个学生自己操作单机,完成各自的作业(题目是随机的,每个学生的题都不一样)。随着学生完成练习,每台学生机的正误率由联网随时反馈到教师机的显示屏上,使教师可以对每个学生的情况进行监视,了解学生练习中的信息,找出薄弱环节。

(六)针对上边练习中的薄弱环节,教师有目的地选择另一些例题发送给学生,并向学生讲解,纠正错误,巩固概念。

二、传统教学与计算机网络教学的比较

1. 形象生动,激发了学生的学习兴趣:

过去做玻一马定律演示实验时,由于大部分学生看不见仪器的读数,教师只能请一个学生上台来读,绝大多数人得不到参与的机会,学生在课堂上较被动,学习兴趣不浓。现在用计算机动态模拟,使每个学生都可看到详细而生动的实验全过程,亲自观察、读数和计算,边思考边讨论,从而提高了学习兴趣,活跃了课堂气氛,学生反映印象深、记得牢。

2. 信息双向传输,反馈迅速:

给学生进行基本训练的题,是从题库中随机选出的,每人题目不一样,造成一个使学生不能抄答案的环境。练习时学生讨论的是解题思路和方法,并通过计

算机进行人机对话。答对了计算机给予肯定,答错了给予提示。学生不会做的题目,可以要求帮助,计算机给出模拟的物理过程,以及简要的文字说明、公式推导和答案。由于学生立即能知道自己答题的对错,因此增加了练习的兴趣,给学生创造了一个生动活泼的学习环境。

在学生练习时,教师可以通过网络系统对每个学生的情况进行监视,了解学生练习中的动态信息。教师一边监视,一边可将学生的情况进行汇总,及时表扬,提高学生学习兴趣。若发现了普遍性的错误,可以用网络将错误发给学生,向学生讲解,当堂就能补漏。

整节课以学生动脑为主,教师讲课为辅,充分发挥了学生的主观能动性。

3. 加强个别化教学,适应各类学生的需要:

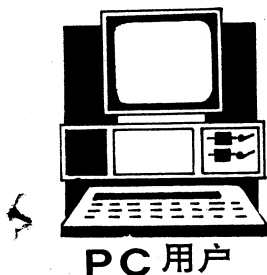
学生在学习由于学习水平不同,总会存在差异。各班的尖子学生,他们理解力强,做题快。过去传统教学只能从中等生出发,开展教学活动,这样使尖子生不能充分发挥他们的才干,压抑了他们的聪明才智。但用计算机辅助教学,随机题库信息量大,可以使尖子生比别人有更多的练习机会。如在玻一马定律的例题中,我们让学生计算,“在开口向上的玻璃管中,用水银封闭了一部分气体,当开口向下放置时,气柱有多长?”有些同学做得较快,于是计算机又提出更深一层的问题,“若玻璃管长为40厘米又如何求?”这些学生讨论得非常热烈,直到下课还在争论着。课后我们向学生了解情况,无论是学习好与差的学生反映均为良好。可见,计算机辅助教学不但能适应中等学生的需要,也有助于尖子生的培养。

4. 增大课堂容量,提高效率,减轻学生负担:

过去讲玻一马定律这一节,做实验、讲解概念后,最多只能做一道例题,大量的练习留在课后完成。现在应用计算机辅助教学,实验及概念讲解后,还能由易到难用标准化方式练习4道题,还加了两道层次不同的例题;学生的问题也及时通过网络而得以发现,并加以纠正。这样在45分钟内,学生既学到了新知识,又培养了能力,课堂上解决了问题,课下负担自然就减轻了许多。

三、结语

实践表明,计算机联网辅助教学不仅保留了传统教学和分立计算机辅助教学的各种优点,而且加强了师生间信息的双向交流,促进教师管理和组织教学,进一步提高学生学习兴趣。我们相信,随着教学方法的进一步完善,计算机联网辅助教学一定会在培养学生分析问题和解决问题的能力上发挥愈来愈大的作用。



也谈 FoxBASE+弹出式菜单的实现方法

呼和浩特内蒙古科技情报研究所(010020) 王瑞华

《电子与电脑》92年第六期刊登的《dBASE Ⅲ和 FoxBASE+环境下实用多窗口弹出式菜单的实现方法》(简称《实用》)一文,为用户提供了方便的弹出菜单的实现方法。由于 FoxBASE+(Rev. 2.10)的速度比 dBASE Ⅲ+快 5.90 倍,而且与 dBASE Ⅲ+100%兼容,所以目前大多数使用 dBASE Ⅲ+的用户纷纷改用 FoxBASE+。但目前熟悉 FoxBASE+的用户不多,所以大多数用户只是把 dBASE Ⅲ的程序直接套入 FoxBASE+中执行。其实, FoxBASE+已经为我们提供了建立弹出式菜单的命令 MENU, MENU TO, @...PROMPT, @(<行>), (<列>)MENU。下面将这些命令作一些介绍,并提供一些实例。

1 建立下拉式菜单命令 MENU

格式: MENU BAR(<数组>), (<数字表达式>)

MENU(<数字表达式 1>), (<数组>), (<数字表达式 2>), (<数字表达式 3>)

READ MENU BAR TO (<变量 1>), (<变量 2>)
[SAVE]

功能: 这个命令建立在顶行的菜单条,并由光标控制键来选择。光标通过的菜单标题将以反象显示,并显示这个标题所包含项目的下拉式菜单。按光标控制键的左右键,则显示相邻标题的下拉式菜单。

实现步骤:

1. 用 DIMENSION 定义顶行菜单棒的数组。
2. 用 DIMENSION 定义与各题目对应的下拉菜单的项目。
3. 用 MENU BAR 使定义的数组进入菜单棒。
4. 用 MENU 将下拉菜单与题目一一对应。
5. 用 READ MENU BAR 激活这个菜单系统。

例 1、资料管理菜单程序

```
set status off
set echo off
set talk off
set message to 24
dimension top(3) && 定义菜单棒数组
top(1)='资料工作'
top(2)='资料借还'
top(3)='资料查询'
dimension zlgz(3) && 定义资料工作菜单数组
zlgz(1)='资料输入'
zlgz(2)='修改内容'
zlgz(3)='删除记录'
dimension jh(2) && 定义资料借还菜单数组
jh(1)='资料借出'
```

jh(2)='资料还入'

dimension zlcx(4) && 定义资料查询菜单数组

zlcx(1)='按书名查'

zlcx(2)='按类号查'

zlcx(3)='按作者查'

zlcx(4)='按内容提要查'

menu bar top,3 && 装配下拉菜单系统

menu 1,zlgz,3,3 && 资料工作菜单装入系统

menu 2,jh,2,2 && 资料借还菜单装入系统

menu 3,zlcx,4,4 && 资料查询菜单装入系统

row=1 && 初始化亮条的位置

col=1

read menu bar to row,col && 激活下拉菜单系统

2 建立上弹菜单命令 @(<行>), (<列>)MENU

格式: @(<座标>)MENU(<一维数组>), (<数字表达式 1>)[, (<数字表达式 2>)][TITLE(<字符表达式>)]

功能: 此命令可在任意位置建立一个上弹菜单。使用可选项 TITLE 可以在顶部显示菜单标题。

实现步骤:

1. 用 DIMENSION 定义一维数组,包括所有的菜单项。

2. @(<行>), (<列>)MENU 指定菜单显示位置,数字表达式 1 是菜单所包括的项目数,表达式 2 是显示在屏幕上的菜单项目数,字符表达式定义菜单标题。

3. 用 READ MENU 激活这个菜单。

例 2、建立上弹菜单程序

```
dimension ch(3,1)
ch(1)='图书采购'
ch(2)='图书编目'
ch(3)='图书流通'
store 0 to cc
@5,0 menu ch,3 title '图书管理系统'
read menu to cc
do case
case cc=1
do cg
case cc=2
do bm
case cc=3
do lt
endcase
```

3 建立条形光亮的菜单命令 @...PROMPT, MENU TO

格式: @(<行>), (<列>)PROMPT(<字符表达式>)[MESSAGE(<字符表达式>)]

MENU TO (内存变量)

功能:这两个命令可以在任何希望的位置显示菜单选择项,可以用光标控制键选择后打回车,或直接键入提示的第一个字符。我们将重点介绍《实用》一文没有用到的 MESSAGE 命令。

实现步骤:

1. 用 @...PROMPT 初始化菜单中的各选择项, MESSAGE 初始化与各项有关的提示信息。
2. MENU TO 将选择项的序号送入内存变量。
3. SET MESSAGE TO (数字表达式)定义提示信息的显示位置。

例 3. 建立条形菜单程序

```
set talk off
@2,1 clea
ch=0
do while ch<>4
@1,10 prompt '资料管理' message space(30)+'输入修改
资料信息'
@1,20 prompt '资料借还' message space(30)+'办理借书
还书手续'
@1,30 prom '资料查询' mess space(30)+'根据书名、作者
查询馆藏资料'
@1,40 prom '结束退出' mess space(30)+'工作结束,返回
```

主菜单'

```
set message to 24
menu to ch
do case
case ch=1
i=1
case ch=2
i=2
case ch=3
i=3
case ch=4
clear
clear all
exit
endcase
i=str(i,1)
do cd&i
enddo
clear
set talk on
```

需注意的两点:1. 如果您采用的是 FoxBASE + (Rev2.00)版本,只能用第三种方式建立菜单。2. 如果是 Rev2.10 版本,则三者都可以使用。

C 语言如何读取 dBASE 的库文件

昆明市北京路 333 号大院自动化站(650051) 李晓华

本文介绍用 C 语言读取 dBASE 数据库文件的方法。

1. dBASE 数据库文件结构:

数据库文件由头信息、结构表、记录集 3 部份顺序组成。

头信息主要包括 dBASE 标识符(dBASE I 与 dBASE II 使用不同的标识符)、记录长度、记录总数、上次更新数据库文件的日期。

结构表主要是描述数据文件的结构,由各字段组成。每个字段描述块有 32 字节。由于 dBASE I 与 dBASE II 所容纳的最多字段不同(dBASE I 为 32 个字段,dBASE II 为 128 个字段),所以其固定分配字节数也不尽相同,结构表由 0X0D 标志结束。

接下来是记录集,它存储数据库文件的全部记录,所占用的空间等于记录长度乘上记录总数,全部数据类型均以 ASCII 码存放。这部分是我们最关心的,特别要注意的是,由于每个记录把第一字节作为“删除标志”,尔后才是存放各个字段的值,所以记录长度应比

各字段长度之和多一个字节。

现以一个人事档案库 RDK.DBF 为例,该库文件的结构为:

姓名	籍贯	性别	出生年月	文化程度	政治面貌
c10	c6	c4	c8	c6	c6

2. 用 C 语言读取 dBASE II 库文件:

首先定义一个对应 RDK.DBF 各字段的数据结

构:

```
/* readdbf.c */
#include <stdio.h>
typedef struct {
    char start[1]; /* 记录的第一字节 */
    char xm[10]; /* 姓名 */
    char jg[6]; /* 籍贯 */
    char xb[4]; /* 性别 */
    char csny[8]; /* 出生年月 */
    char whcd[6]; /* 文化程度 */
    char zzmm[6]; /* 政治面貌 */
```



```

} rdk;
main(){
    FILE *filedbf; /* dBASE 数据库文件名 */
    rdk *buff; /* 存放一条记录的缓冲区 */
    char xml[11],jg1[7],xb1[5],csny1[9]
    char whcd1[7],zzmm1[7]; /* 存放各字段 */
    filedbf = fopen("RDK.DBF","rb"); /* 打开 dBASE 数据库文件 */
    if(filedbf == '\0'){ /* 未找到.DBF文件 */
        puts("can't open .DBF file!");
        exit(1);
    }
    buff = (rdk *)malloc(sizeof(rdk)); /* 申请内存空间 */
    if(buff == NULL){ /* 无足够的内存空间 */
        puts("out of memory");
        exit(0);
    }
    while(fgetc(filedbf) != 0x0d){ /* 找记录的开始 */
        fgetc(filedbf); /* 跳一个记录 */
        puts("姓名 籍贯 性别 出生年月 文化程度 政治面貌");
        while(fread(buff,sizeof(rdk),1,filedbf) != NULL){
            addeof(buff->xm,xm1,10); /* 在各字段尾加'\0' */
            addeof(buff->jg,jg1,6);
            addeof(buff->xb,xb1,4);
            addeof(buff->csny,csny1,8);
            addeof(buff->jg,jg1,4);
            addeof(buff->whcd,whcd1,6);
            addeof(buff->zzmm,zzmm1,6);
            printf("%s%s %s%s %s %s\n",xm1,jg1,xb1,csny1,whcd1,zzmm1);
        }
    }
    int addeof(char *c1,char *c2,int n) /* 在各字段尾加'\0'的函数 */
    {
        while(n-- != 0) *c2++ = *c1++;
        *c2 = 0;
    }
}

```

说明:1. 读取文件时,应首先找到库文件结构表的结束标志 0X0D。

2. 由于文件的每条记录开始,都有一个字节作“删除标志”,所以读取时应过滤该字节。

3. 用 C 语言显示字符串时,需要有一个标志 '\0',所以,.DBF 库的每个字段末应加上 '\0' 标志后,才能正确显示。

4. 如库的字段为数据类型,且要进行数字运算时,需要用 atoi(s)函数把字符串转换成数字。

如何实现题库系统的图文并貌

天津师范大学计算机系 裴伟东

如何实现题库系统的图文并貌,是一直没有得到较好解决的一个问题。过去大多是在建立题库的同时,造出大量的图形符号,并利用它们拼成某种图形存放在题库中。这种方法的主要缺点是工作量太大,需要较多的人力和较长的时间才能完成。下面,根据我们在开发“离散数学题库系统”过程中的一些体会,向读者介绍一种简单实用的方法,供大家参考,并抛砖引玉。

一、绘图工具的设计

为实现题库系统的图文并貌,我们用 FoxBASE+2.00 和 2.13H 系统的特殊显示功能,首先设计了一个绘图程序 ht.prg,其功能、绘图方式及绘图原理如下:

1. 功能

ht.prg 的主要功能是:

- (1) 绘制带有节点的树型图(有向图或无向图)。
- (2) 绘制表格、矩阵型图(有向图或无向图)。
- (3) 把绘好的图形特征信息存储在图形特征库中。

2. 绘图方式

ht.prg 用 FoxBASE 的绘图语句将屏幕划分成三个主要的窗口,它们是:

- (1) 绘图窗口。该窗口规定了绘制图形的大小范

围,所使用的绘图语句只能把图形绘制在该窗口内。在整个图形的绘制过程中,从头到尾每一步的绘制图象都显示在该窗口内,以待用户给出是否满意的信息。

(2) 提示窗口。该窗口给出每一步要绘制的图象内容,包括节点、连线、数据(字母、汉字等也属数据内容)、它们的位置、连线的方向和特征等信息,待用户选择后,该步所绘制的图象即显示在绘图窗口中。

(3) 操作窗口。该窗口主要是选择提示窗口上的信息,并在每一步所绘制的图象显示在绘图窗口之后,询问用户是否满意、是否要修改、整个图形是否已画完等操作信息。

用户通过以上三个窗口的反复操作,最后将一个理想的图形绘制在绘图窗口内。

3. 图形的存储方式

绘图程序 ht.prg 的另一个功能还在于它能存储一个图形的全部特征信息,为此,我们建立了一个存储图形特征信息的特征库 TU.DBF。考虑到各种图形内部有的有数据,有的无数据,数据的多少又和图形的复杂程度没有必然的联系等,为了节省特征库 TU.DBF 的存储空间,我们另行建立了一个数据库 DATA.DBF

用以专门存放图形中的数据;DATA. BF 通过 TU. DBF 中的一个字段与 TU. DBF 发生一一对应的关系,这两个库的结构如图 1.1 和图 1.2

Field	Field Name	Type	Width	Dec
1	X1	Numeric	2	
2	Y1	Numeric	2	
3	UJ1	Character	2	

图 1.1 DATA. DBF 库结构

图 1.1 中,X1,Y1 记录图形当前数据的位置,是文本方式下 X,Y 的坐标;UJ1 记录当前数据的内容。

Field	Field Name	Type	Width	Dec
1	X	Numeric	2	
2	Y	Numeric	2	
3	L1	Numeric	2	
4	L2	Numeric	2	
5	FX1	Character	20	
6	FX2	Character	20	
7	FX3	Character	20	
8	FX4	Character	20	
9	FX5	Character	20	
10	FX6	Character	20	
11	FX7	Character	20	
12	FX8	Character	20	
13	U1	Character	3	
14	V1	Character	3	
15	U2	Character	3	
16	V2	Character	3	
17	U3	Character	3	
18	V3	Character	3	
19	U4	Character	3	
20	V4	Character	3	
21	UJ	Numeric	1	

图 1.2 TU. DBF 库结构

图 1.2 中,X,Y 记录节点或连线的位置;是文本方式下 X,Y 的坐标;L1,L2 记录与当前节点或连线位置有关的点;FX1-FX8 记录当前连线的方向特征,它们分别为上、下、左、左上、左下、右、右上、右下等;U1,V1~U4,V4 为各种方向的位置点;UJ 是图形中有无数据的标记,当 UJ=1 时,表示该图形中有数据,数据内容存储在相对应的 DATA. DBF 中,当 UJ=0 时,表示该图形中没有数据。

以上两库中各字段的意义,包含了一个图形应具备的各种特征信息。

4. ht. prg 的程序原理

简单地说,ht. prg 就是从提示窗口获得图形特征信息,再把该信息所代表的图象用绘图语句绘制在绘图窗口,如果用户满意,则将该特征信息存储在 TU. DBF 和 DATA. DBF 中,整个图形绘制完毕时,描绘该图形的全部特征信息也就存储在固定的特征库里了。

鉴于树型图与表格、矩阵型图在绘制原理上的一致性,这里只给出 ht. prg 关于树型图的绘制原理。

(1)第一步是把一个树型图的全部节点画好;第二步是连节点之间的连线并画好连线的方向。具体过程是从第一个节点连起,直到最后一个节点完毕,且这个过程可重复一遍;第三步是填好图形中的数据。

(2)程序中并没有存储节点的形状,这是因为树型图的节点形状不变,我们统一规定了它们的形状都为“。”。例如,要在坐标点(x,y)处画一个节点,可用语句 @x,y say“。”实现。可见,节点形状并不需要存储。

(3)连线和连线方向的画法是通过用 FoxBASE 语句和 2.13H 系统的特殊显示功能来实现的。例如,要把坐标(x₁,y₁)处的节点与坐标(x₂,y₂)处的节点连起来,可用下面两条语句来实现:

```
@1,1 say chr(14)+"Dx1,y1"]
```

```
@1,1 say chr(14)+"Lx2,y2"]
```

其中:(x₁,y₁)和(x₂,y₂)必须是图形方式下的坐标,它们与文本方式下的坐标可相互转换。例如,在 EGA-26 行显示模式下,设文本坐标为(x,y),则其图形坐标(x₁,y₁)的值可由下式计算:

$$x_1 = \frac{350}{26}x \approx 13.46x, y_1 = \frac{640}{79}y \approx 8.10y$$

二、图形显示工具的设计

由于 ht. prg 将绘好的图形以图形特征信息存储在特征库文件中,当调用图形时,就必须把其特征信息还原成真正的图形,为此,我们利用类似前面的原理,设计了一个图形显示程序 xt. prg,其功能和原理如下:

1. 功能 显示由 ht. prg 绘制过的图形。

2. 原理 xt. prg 原理简介如下。

ht. prg 本身就包含了一个绘图过程,只不过它是从提示窗口获得一个图象信息,再把该信息代表的图象绘制出来,而 xt. prg 的不同之处在于,它不是从提示窗口获得图象信息,而是从图形特征库中获得信息,再把该信息所代表的图象绘制出来。两个程序中所使用的绘图语句都是一样的。

三、题库中如何存取图形

由 ht. prg 的绘图原理我们知道,ht. prg 把绘制过的图形特征存放在固定的特征库 TU. DBF 和 DATA. DBF 中,它们目前还没有和题库中的题目产生一一对应的关系,那么,题库中的某个题目与它的图形怎样一一对应呢?图形的存取方式又是怎样的呢?下面,我们就这两个问题分别加以讨论。

(1)题库中题、图对应关系的实现

我们以一个具体的例子,来说明在录入题目时,某个题目与它的图形怎样产生一一对应的关系。

设“离散数学”中关于“树”部分的题目存放在 LS5. DBF 中,为方便说明起见,我们把该库的结构简化为 TM、BZ、TU 三个字段。其中:

TM——题目中除图以外其它必备字段的总和;
BZ——标志字段;当 BZ=1 时,表示当前题目有图;
TU——当前图形的文件名字段,其中文件名代表存储该图形特征信息的特征库文件名。

当录入这部分题目到 LS5. DBF 中时,某个题目与其图形的对应关系及存图方式流程如下:

用计算机模拟“卡拉—OK”呢?我用 BASIC 编制了一个程序,完成了音乐演奏和歌词显示效果。程序执行后,屏幕显示电子琴的演奏过程,同时随音乐依次显示相应的歌词。可同时输入9首歌曲供选择。该程序已在286微机中通过。本程序只输入了《在水一方》一首歌。

程序说明

第2至270行,电子琴键盘显示。

第490至620行为演奏子程序,可完成音乐演奏、电子琴演奏过程显示及歌词显示功能。

第370、380行,计算发音频率。

第630至769行,选择歌曲。键入A—I可选择9首不同的歌曲,按ESC键退出。

第1000行以后,歌曲置数语句。歌曲A从1600行开始置数;歌曲B从1700行开始置数;歌曲C从1800行开始置数;歌曲D从1000行开始置数;歌曲E从1100行开始置数;歌曲F从1200行开始置数;歌曲G从1300行开始置数;歌曲H从1400行开始置数;歌曲I从1500行开始置数。

程序中第14、220、260、610、620的空格字串为两个字符;170~190句字串中的空格间隔为16字符。

```

2  COLOR 15,0,0
5  CLS;COLOR 0,12
10 FOR I=5 TO 74;FOR J=0 TO 1
11  LOCATE 1+J*8,I;PRINT " "
12  NEXT J;NEXT I
13 FOR I=0 TO 1;FOR J=2 TO 8
14  LOCATE J,5+I*68;PRINT " "
15  NEXT J;NEXT I
150 COLOR 15,0
170 LOCATE 11,15;PRINT "A—B—C—"
180 LOCATE 12,15;PRINT "D—E—F—在水一方"
190 LOCATE 13,15;PRINT "G—H—I—"
200 COLOR 0,15
205 FOR J=2 TO 8
210 FOR I=0 TO 15
220 LOCATE J,7+I*4;PRINT "|";" "
230 NEXT I
231 PRINT "|"
232 NEXT J
235 COLOR 15,0
240 FOR I=0 TO 12;FOR J=2 TO 5
250 IF I=2 OR I=6 OR I=9 OR I=13 THEN 270
260 LOCATE J,15+I*4;PRINT " "
270 NEXT J;NEXT I
275 COLOR 15,1,1
360 DIM M(88),O(70)
370 FOR I=7 TO 88;M(I)=36.8*(2^(1/12))^(I-6);
NEXT I
380 FOR I=0 TO 6;M(I)=32767;NEXT I
390 O(0)=0
400 O(39)=5;O(40)=7;O(41)=8;O(42)=9
410 O(43)=10;O(44)=11;O(45)=13;O(46)=14
420 O(47)=15;O(48)=16;O(49)=17;O(50)=18
430 O(51)=19;O(52)=21;O(53)=22;O(54)=23

```

```

440 O(55)=24;O(56)=25;O(57)=27;O(58)=28
450 O(59)=29;O(60)=30;O(61)=31;O(62)=32
460 O(63)=33;O(64)=35;O(65)=36;O(66)=37
470 O(67)=38;O(68)=39;O(69)=40;O(70)=42
480 GOTO 630
490 READ J,K,L
500 CMD$=INKEY$;IF CMD$="" THEN 540
510 IF CMD$=CHR$(27) THEN RETURN
540 IF J=-1 THEN RETURN
541 IF J<>-3 THEN 550
542 COLOR 14,4;LOCATE 16,6
543 PRINT " "
544 LOCATE 17,6
545 PRINT " "
546 LL=0;READ SS$;LOCATE 16,14;PRINT SS$
547 GOTO 490
550 Q=O(J)
555 LL=LL+L;COLOR 14,4;LOCATE 17,12+2*LL;
PRINT ">"
560 IF J>64 OR J<39 THEN 590
570 IF INT(Q/2)-Q/2<>0 THEN COLOR 14,4;LO-
CATE 7,2*Q-1;PRINT "◎";GOTO 590
580 COLOR 14,4;LOCATE 3,2*Q-1;PRINT "◎";
590 SOUND M(J),K;IF J=0 AND K=1 THEN 600;
'SKIP NEXT FOR STACCATTO
595 SOUND 32767,1
600 IF J>64 OR J<39 THEN 490
610 IF INT(Q/2)-Q/2=0 THEN COLOR 15,0;LO-
CATE 3,2*Q-1;PRINT " ";GOTO 490
620 COLOR 0,15;LOCATE 7,2*Q-1;PRINT " ";
GOTO 490
630 LOCATE 15,15;PRINT "ENTER SELECTION ==
>";
631 COLOR 14,4;LOCATE 16,6
632 PRITN " "
633 LOCATE 17,6
634 PRINT " "
650 IF INKEY$<>" " THEN 650
660 CMD$=INKEY$;IF CMD$="" THEN 660
670 IF CMD$=CHR$(27) THEN 850
685 IF CMD$="A" OR CMD$="a" THEN RESTORE
1600;GOTO 770
690 IF CMD$="B" OR CMD$="b" THEN RESTORE
1700;GOTO 770
700 IF CMD$="C" OR CMD$="c" THEN RESTORE
1800;GOTO 770
710 IF CMD$="D" OR CMD$="d" THEN RESTORE
1000;GOTO 770
720 IF CMD$="E" OR CMD$="e" THEN RESTORE
1100;GOTO 770
730 IF CMD$="F" OR CMD$="f" THEN RESTORE
1200;GOTO 770
740 IF CMD$="G" OR CMD$="g" THEN RESTORE
1300;GOTO 770
750 IF CMD$="H" OR CMD$="h" THEN RESTORE
1400;GOTO 770

```

```

755 IF CMD$ = "I" OR CMD$ = "i" THEN RESTORE
    1500,GOTO 770
769 GOTO 630
770 READ D
800 READ S$:LOCATE 15,35
805 COLOR 14,4:PRINT S$:COLOR 0,7
810 GOSUB 490
820 LOCATE 15,35:COLOR 15,0 PRINT"
"
830 GOTO 630
840 END
1200 DATA -2,"在水一方"
1205 DATA -3,0,0"绿草苍苍,白雾茫茫,有位佳人在水
    一方。"
1206 DATA 47,6,1,49,6,0,52,36,1,59,6,1,61,6,0,56,
    36,1,49,6,2,52,6,0,56,6,0,54,36,1,56,6,1,61,6,
    0
1207 DATA 56,6,0,59,36,1,59,12,2,61,36,1,63,6,1,
    61,6,0,59,6,0,56,36,1,47,6,1,59,3,0
1208 DATA 56,3,0,54,36,1,49,6,1,51,6,0,52,36,1
1215 DATA -3,0,0"绿草萋萋,白雾迷离,有位佳人靠水
    而居。"
1216 DATA 47,6,1,49,6,0,52,36,1,59,6,1,61,6,0,56,
    36,1,49,6,2,52,6,0,56,6,0,54,36,1,56,6,1,61,6,
    0
1217 DATA 56,6,0,59,36,1,59,8,2,61,36,1,63,6,1,61,
    6,0,59,6,0,56,36,1,47,6,1,59,3,0
1218 DATA 56,3,0,54,36,1,49,6,1,51,6,0,52,48,1
1220 DATA -3,0,0"我愿逆流而上,依偎在她身旁,无奈
    前有险滩,道路又远又长"
1222 DATA 64,18,1,63,6,1,61,6,1,63,6,0,64,12,1,
    56,6,1,57,6,0,59,36,1
1224 DATA 61,12,2,59,6,1,57,6,0,56,12,1,49,6,1,
    56,6,1,54,36,1
1226 DATA 56,6,2,57,6,0,59,18,1,61,6,1,59,12,1,
    57,6,1,56,6,0,54,36,1
1228 DATA 54,6,2,56,6,0,57,18,1,64,6,1,63,12,1,
    54,6,1,61,6,0,59,48,1
1230 DATA -3,0,0"我愿顺流而下,找寻她的方向,却见
    依稀仿佛,她在水的中央"
1232 DATA 64,18,1,63,6,1,61,6,1,63,6,0,64,12,1,
    56,6,1,57,6,0,59,36,1
1234 DATA 61,12,2,59,6,1,57,6,0,56,12,1,49,6,1,
    56,6,1,54,36,1
1236 DATA 56,6,2,57,6,0,59,18,1,61,6,1,59,12,1,
    57,6,1,56,6,0,54,36,1
1238 DATA 54,6,2,56,6,0,57,12,1,63,6,1,61,6,0,59,
    12,1,51,6,1,54,6,0,52,48,1
1240 DATA -3,0,0"我愿逆流而上,与她轻言细语,无奈
    前有险滩,道路曲折无已"
1242 DATA 64,18,1,63,6,1,61,6,1,63,6,0,64,12,1,
    56,6,1,57,6,0,59,36,1
1244 DATA 61,12,2,59,6,1,6,57,6,0,56,12,1,49,6,1,
    56,6,1,54,36,1
1246 DATA 56,6,2,57,6,0,59,18,1,61,6,1,59,12,1,
    57,6,1,56,6,0,54,36,1

```

```

1248 DATA 54,6,2,56,6,0,57,18,1,64,6,1,63,12,1,
    54,6,1,61,6,0,59,48,1
1250 DATA -3,0,0,"我愿顺流而下,找寻她的踪迹,却见
    依稀仿佛,她在水中伫立"
1252 DATA 64,18,1,63,6,1,61,6,1,63,6,0,64,12,1,
    56,6,1,57,6,0,59,36,1
1254 DATA 61,12,2,59,6,1,57,6,0,56,12,1,49,6,1,
    56,6,1,54,36,1
1256 DATA 56,6,2,57,6,0,59,18,1,61,6,1,59,12,1,
    57,6,1,56,6,0,54,36,1
1258 DATA 54,6,2,56,6,0,57,12,1,63,6,1,61,6,0,59,
    12,1,51,6,1,54,6,0,52,48,1
1260 DATA -1,-1,-1
1600 DATA -2," "
1611 DATA 52,20,1,43,20,1,44,20,1,45,20,1,46,20,
    1,47,20,1,48,20,1,49,20,1,50,20,1,51,20,1,52,
    20,1,53,20,1,54,20,1,55,20,1,56,20,1,57,20,1,
    58,20,1,59,20,1
1612 DATA 60,20,1,61,20,1,62,20,1,63,20,1,64,20,
    1,65,20,1
1613 DATA -1,-1,-1

```

实现1.2MB软磁盘之间直 接拷贝简法

河南省人民银行计算中心(450002) 何崇乐

近读《电子与电脑》92年第4期上的《在单台驱动器上如何作软盘备份》(下称“杨文”),使我想起《软件报》92年第6期上的《一种快速简便的整盘文件拷贝法》(下称“软文”)及去年《中国金融电脑》(9/91)上的《怎样实现1.2MB软磁盘之间直接拷贝》(下称“融文”);三文虽说用法不同,但似均为舍近求远,因此,愿以拙见就教于同行。

首先,不论在DOS 2.X(!)还是DOS 3.X下,即使对于双软驱机,也均可按单软驱机的盘拷法进行整盘拷贝。(此点DOS手册上本已有阐述;或如“杨文”所说:“其实,DOS的设计者为我们考虑了这个问题……”)

只是,若A驱为高密驱动器,因DOS 2.X管辖权所限,使DISKCOPY命令无法正确执行;换言之,符合我们要求的,只能是使用DOS 3.X操作系统。

其次,在高密A驱上由1.2MB盘直接(真正!)拷贝1.2MB盘的方法如下:

在DISKCOPY.COM所在目录下(否则,得指定

PATH),直接键入

DISKCOPY A: A:<CR>

即可;且其间的换盘信息和换盘操作,与在单软驱时的完全相同。

须指出的是,与 DOS 2.X 下的 DISKCOPY 操作不同,在 DOS 3.X 下,驱动器的指定既不能全部缺省,也不能部分缺省,而是缺一不可——必如上述方可。

以此与上述三文所用方法相较,可以看出:

“软文”所述方法,除了插盘次数要少些外,则无论从省时,还是从节省相对贵重的资源——硬盘——来看,均不理想。

而“融文”所说的方法,实际上既不“直接”,又属多此一举,更不理想。

至于“杨文”的方法:除非你必须用 COPY 命令来实现(个别文件的)复制,此时只好如“杨文”所述费事(建/补两个.SYS 文件)外;只要想以 DISKCOPY 来实现复制(哪怕此后再删去无关文件),则显然以本法为简!何况,“杨文”的原意,也似在(全盘)“备份”。

正因如此,聊成此文,以对不知、不详“直接拷贝”法者有所裨益。

兼容机怎样运行 BASICA.COM

西苍梧佛子冲矿计算室(543100) 邓文超

因 BASICA.COM 需 ROMBASIC 支持,而一般兼容机都没插有容量为32K 的 ROMBASIC 芯片,所以多数兼容机不能运行 PC-DOS 的 BASICA.COM,而只能运行 MS-DOS 的 GWBASIC.EXE。极大多数 BASIC 语言教科书均以 PC-DOS 的 BASICA 为范例(最明显的是功能键的定义),况且 BASICA.COM 具有调用 DOS 命令(SHELL“命令”)等实用功能,因此多数用户更乐意使用 BASICA.COM。

下面介绍一个让兼容机也能运行 BASICA.COM 的方法,此方法使用了 DOS 的重定向功能,其优点在于每次运行 BASICA.COM 时只要打入 BAS 即可,无需多余的操作,就象在 IBM 机上使用 BASICA 一样简便。以下是一次性的准备工作,共六点:

1. 准备一张空软盘,将 DEBUG.COM 文件和 PC-

TOOLS.EXE 文件拷贝到该软盘上。并参照 DEBUG 的命令格式,在盘上建立一个借助 DEBUG 读取 ROMBASIC 参数的命令集文件,文件取名为 READROM.TXT:

```
A> TYPE READROM.TXT
M F600:0 L 8000 100
R CX
8000
N A:ROMBAS.DAT
W
Q
```

2. 将此软盘插入 IBM 原装机中,在 A>下打入 DEBUG< READROM.TXT,几秒钟后即可在软盘上生成一个名为 ROMBAS.DAT 的含有 ROMBASIC 参数的数据文件。

3. 将 BASICA.COM 拷贝到软盘上。

4. 启动 PCTools,选择软盘的 BASICA.COM 文件,按 F 键进入查找功能,按 F1键变换查找方式为找 16进制码,输入 00F6并回车,开始查找。找到后按 E 键对 00F6进行编辑,改为 0070,然后按 F5键认可,按回车键返回,再按 G 键继续查找,找到再修改,直到全部 00F6改为 0070为止。退出 PCTools。

5. 参照 DEBUG 的命令格式建立以下装入 ROM-BASIC 参数的命令集文件 BAS.TXT 于软盘上:

```
A>TYPE BAS.TXT
N A:ROMBAS.DAT
L
M 100 L8000 7000:0
Q
```

6. 再在软盘上建立以下启动 BASICA.COM 的批命令处理文件 BAS.BAT:

```
A>TYPE BAS.BAT
A:DEBUG <A:BAS.TXT
A:BASICA
```

以上 DEBUG 后跟的“<”符号为 DOS 的重定向符号。

至此,准备工作已完成,软盘上已含有:PC-TOOLS.EXE、READROM.TXT、DEBUG.COM、BAS.TXT ROMBAS.DAT、BASICA.COM(已修改段地址 F600为7000)、BAS.BAT 共七个文件,其中后五个文件是在兼容机上运行 BASICA.COM 所必需的。

当要运行 BASICA.COM 时,插软盘入兼容机,打入 BAS 并回车即可进入 BASICA。

若要使用带执行文件名的方式进入 BASICA 的话,不妨按如下步骤进行:

```
A) DEBUG<BAS.TXT
A)BASICA 文件名(CR)
```

即先装入 ROMBASIC(每次开机装入一次即可),然后再象平常那样启动 BASICA.COM,非常方便。

香港病毒的消除和防范

南京东南大学四系四四教研组(210018) 孟桥

在一台计算机使用过程中,我们发现拒不执行打印命令的情况。经过仔细检查,排除了硬件故障的可能性。再用 CHKDSK 命令检查,发现可用内存比平时少了1K。于是怀疑有病毒作祟。用了几种病毒扫描软件均未发现异常。后用 CPAV 软件,查出其中有“香港病毒”(Hong Kong virus)。但是该软件因为缺乏对该病毒的分析资料,所以不能解除硬盘上的病毒体。

为了消除硬盘上的“香港病毒”,我们仔细分析了该病毒的机制和结构,并找出了一种较为简单的消除并防范该病毒的方法。

一、病毒的作用机制和危害

“香港病毒”载体存在于硬盘或软盘的绝对扇区0中,对于软盘,这里应该是引导程序所在之处。病毒在感染时将原来该扇区的内容存入第39道8区1头中。如该盘是360K 软盘,则这是最后一个逻辑扇区,一般除非软盘已用满,否则不会用到它。但是对于1.2M、1.44M 和720K 的软盘而言,它正处于中间位置,极易破坏软盘上的数据。对于硬盘,此处放的是主引导程序,此时病毒体用其自身替代主引导程序,覆盖了原来的内容。此是其危害之一。

病毒体本身也具有一定的分区引导功能,但比较简单,缺乏对引导分区的检验、分区表的检查、系统错误提示、ROMBASIC 引导等等功能。所以一旦系统不正常就会引起死机,而不会给用户以任何提示信息。此其危害之二。

当计算机开机启动时,病毒载体替代原来的主引导程序(硬盘启动时)或引导程序(软盘启动时)而首先进入系统内存并执行。它将病毒传播程序链接于DOS系统的中断INT 13H(磁盘I/O)中,将自身拷贝到内存高端,并通过改变BIOS参数区使可用内存减少1K字节,对病毒体进行保护。此其危害之三。

病毒体链接于INT 13H 上后,每当系统进行磁盘I/O时,首先执行病毒传播程序,对被读写的磁盘进行感染,既传播了病毒,也降低了程序运行速度。对硬盘的感染则是在用有毒软盘启动时进行。此其危害之四。

当病毒载体把病毒传播程序链接于中断INT 13H 之后,便使其内部计数器加一,同时写回磁盘。该计数器记录了病毒感染后机器被启动的次数。当计数器的值大于512时,病毒载体通过修改BIOS参数对打印口PRN1、串行通信口COM1进行“软锁”,使打印机、串行口操作不能正常进行。例如:当使用打印机时,屏幕上出现“No paper...”等字样。病毒的发作具有一定的潜伏期,故发作时往往已进行了一定范围的传染,

而发作时的症状极易被判断为硬件故障,给计算机的维修带来一定的困难。此其危害之五。

综上所述,“香港病毒”侵害计算机的最终目的在于“软锁”部分硬件外设,给使用者带来一些麻烦。但它不进攻击计算机的数据系统,应该算是良性病毒。

二、“香港病毒”的消除和防范

目前一般常用的防毒软件都无法消除“香港病毒”。由于它存在于系统的引导区中,可以先将所有硬盘中文件进行备份,再重新格式化(包括低级格式化)硬盘,将文件重新拷入系统,最后消除(格式化)数据备份软盘(因为此时软盘上一定感染了病毒)。但是这样做工作量太大。这里介绍一个简单的方法,可以有效地用以消除和防范“香港病毒”。

对于软盘而言,消除该病毒的方法非常简单。首先判断是否感染了病毒。该病毒的特征是在偏移089h和08Ah处的码字为B1和90。判别软盘有病毒以后,只要将被隐藏的正常BOOT区写回即可。下面的一段程序KILLHKA.COM就是用于消除A盘的病毒的。它首先判断A盘是否有香港病毒。若有,则进行消毒:

```
C>DEBUG
-A100
XXXX:0100 MOV AX,0
XXXX:0103 INT 13
XXXX:0105 MOV AX,0201
XXXX:0108 MOV BX,0400
XXXX:010B MOV CX,1
XXXX:010E MOV DX,0
XXXX:0111 INT 13
XXXX:0113 JNB 117
XXXX:0115 INT 20
XXXX:0117 MOV AX,[0489]
XXXX:011A CMP AX,90B1
XXXX:011D JNZ 0115
XXXX:011F MOV AX,0201
XXXX:0122 MOV BX,0400
XXXX:0125 MOV CX,2708
XXXX:0128 MOV DX,0100
XXXX:012B INT 13
XXXX:012D MOV AX,0301
XXXX:0130 MOV BX,0200
XXXX:0133 MOV CX,1
XXXX:0136 MOV DX,0
XXXX:0139 INT 13
XXXX:013B INT 20
XXXX:013D
-RCX
CX 0000
:100
-RBX
BX 0000
:0
-NKILLHKA.COM
-W100
Writing 100 bytes.
```

-Q

以后只要在 DOS 提示符下执行 KILLHKA 就自动完成对 A 盘的查毒和解毒。

对于硬盘的消毒过程较之软盘略为复杂一些。先在正常、无病毒感染的机器上通过 DEBUG 程序建立一个消病毒程序 KILLHKH.COM, 具体过程如下:

(1) 拷贝正常的主引导区至内存中(偏移地址 0200H)

-A100

XXXX:0100 MOV AX,0201

XXXX:0103 MOV BX,200

XXXX:0106 MOV CX,1

XXXX:0109 MOV DX,80

XXXX:010B INT 13

XXXX:010D INT 20

XXXX:010E

-G=100

Program terminated normally

(2) 编制消病毒程序:

-A100

XXXX:0100 MOV AX,0

XXXX:0103 INT 13

XXXX:0105 MOV AX,0201

XXXX:0108 MOV BX,0400

XXXX:010B MOV CX,1

XXXX:010E MOV DX,80

XXXX:0111 INT 13

XXXX:0113 JNB 117

XXXX:0115 INT 20

XXXX:0117 MOV AX,[0489]

XXXX:011A CMP AX,90B1

XXXX:011D JNZ 0115

XXXX:011F MOV SI,0403

XXXX:0122 MOV DI,0203

XXXX:0125 MOV CX,8

XXXX:0128 CLD

XXXX:0129 REPZ

XXXX:012A MOVSB

XXXX:012B MOV SI,0570

XXXX:012E MOV DI,0370

XXXX:0131 MOV CL,90

XXXX:0133 REPZ

XXXX:0134 MOVSB

XXXX:0135 MOV AX,0301

XXXX:0138 MOV BX,0200

XXXX:013B MOV CX,1

XXXX:013E MOV DX,80

XXXX:0141 INT 13

XXXX:0143 JMP FFFF:0000

XXXX:0148

-RCX

CX XXXX

:400

RBX

BX XXXX

:0

-NKILLHKH.COM

-W100

Writing 400 bytes.

-Q

此时,一个专用于消除硬盘的“香港病毒”的软件就形成了。把该文件拷入被感染的机器内并执行 KILLHKH,若该机未感染香港病毒,则程序正常退出,不作任何动作。若发现香港病毒,则自动消除,并进行重新启动,从而制止病毒的再次感染。

该消毒程序无须用健康系统盘重新启动系统,所以使用非常方便。它可放在系统自启动批文件 AUTOEXEC.BAT 中,起到防毒、消毒的作用。

~~~~~

## 谈数据文件“死而复生”的技巧

兰州军区司令部作战部(730000) 李瑛彬

在计算机世界里,一个个数据文件三天两头被判为“死刑”,绝非偶然。但也有一些文件被误判为“死刑”,那简直是软件研制过程中时间、人力、经费、成果的极大损失。下面介绍几种文件被删或突然丢失后被挽回的方法,供参考。

### 一、文件或目录被损坏后的恢复法

使用 DOS 命令 RECOVER.COM 进行恢复。恢复 A 软盘批量文件:

C>recover a:✓

恢复 A 软盘的 LYB.WSF 单个文件:

C>recover a:Lyb.wsf✓

### 二、文件被误删后的恢复法

使用 PCTools 工具软件中的 U 命令,步骤如下:

C>pctools✓

按 F10 键选择即要恢复的软盘驱动器,按 F3 键后用 U 键命令,此时系统提示所有被删除的文件,文件打头的字符全部为?号。如,恢复 A 软盘的 LYB.WSF 文件,系统提示为?YB.WSF。可将光标移至这个文件上,将?号改为 L,再打 G 键命令,文件马上就被恢复,然后按 Esc 键返回 DOS 系统。

### 三、写盘过程的文件挽救法

笔者正在往 A 软盘写数据文件 909XT,突然别人将盘拿走,出现以下现象。

Not ready error reading drive A

Abort,Retry,Ignore,Fail?

出现这种情况时,千万不能乱击键或进行热启动。待把那张软盘插入 A 驱动器后,按 R 键即可将文件全部存入 A 软盘。

### 四、数据文件未做写处理丢失的恢复法

上述三种文件的特点是都有数据基础,因为它曾



驻留过磁盘。但有时在编写程序或打印文字材料时,在写盘失败后进行第三种方法处理仍不奏效,或干脆未做访问软驱的动作,直接返回 DOS,造成文件丢失的现象更为可悲。如出现这种情况,笔者在此提出严肃忠告,不要运行其它程序,更不能关机或热启动。因为此时文件内容仍驻留在内存之中。如果运行其它程序,会造成因其它程序占用内存较大空间而冲掉内存中的数据文件,重新关机或热启动会造成数据文件干脆丢失,而无法挽救。此时可用 DEBUG.COM 文件进行恢复。现假设本文未做访问软盘驱动器而突然丢失的情况为例,对其进行恢复的方法。

C>debug

-s 0 ffff 谈数据文件“死而复生”的技巧,搜索文件首地址

-66A0:5AB8 ,显示首地址

-na,909xt ,将恢复文件存入 A 盘

-rcx

-CX 0000 ,显示寄存器

,7190 ,根据文件长度给出偏移量

-w5ab8 ,存入首地址

Writing 7190 bytes

-q ,退出

用 DEBUG.COM 修改完之后,再用汉字编辑程序调入 A:909XT 文件就会发现文件完好无损。

## 数据自动存盘程序

四川大学数学系(610064) 汪海波

### 一、用途

(一)当有文字编辑软件进行文字数据录入时,突然的停电,将使您的工作前功尽弃。用手动存盘可以避免一些损失,但一来操作麻烦,二来由于疏忽不能及时存盘的事常常不幸地发生;而市场上出售的不间断电源价格昂贵。笔者编写的这个程序较好地解决了这个问题。本程序对录入的文字数据具有自动存盘功能,在一般情况下,几乎不影响用户的正常操作,而一旦停电或由于某种原因造成微机死锁,又未能及时存盘,本程序将使您很方便地恢复刚录入的文字数据。

(二)具有“黑盒子”的功能。将本程序放在硬盘上,在自动批文件里加上一两条命令,就具有对用户上机操作进行自动监视记录的功能。这对于微机管理员及时查出系统故障的原因,或教师了解学生上机作业情况都有很大的作用。

### 二、原理

本程序通过修改 INT 16H 键盘管理中断向量,指向新的中断程序,对键盘输入进行记录。当输入字符数达到64字节(可改成其它的数字)时,就自动存盘一次。键盘输入信息记录在数据文件 SAVE.DAT 中。

编写程序时,使用了下面一些技巧:

1)每次存盘时,移动文件读写指针到文件尾部,只存入64个字节。这在一瞬间即可完成,因此对用户的正常操作几乎没有影响。

2)每次启动本程序,自动将上次的记录文件 SAVE.DAT 改名为 SAVE.BAK,并重新建立新的记录文件 SAVE.DAT,从而保证 SAVE.DAT 中记录的是本次启动后的键盘输入。上次启动后的键盘输入可在 SAVE.BAK 中查到。

### 三、程序

本程序用汇编语言写成,经编译后的执行文件 AUTOSAVE.EXE 只有1K 字节。

将文字编辑软件盘插入 A 驱动器,将含有编译程序 MASM.EXE 及连接程序 LINK.EXE 的软盘插入 B 驱动器,按下列步骤操作:

1)在 DOS 提示符下打 WS<CR>(CR 表示按回车键,下同)选择 N(编辑非文书文件)项,取名 B:AUTOSAVE.ASM,然后按下面的源程序清单输入计算机。输完后,打 Ctrl+KX,存盘退出。

2)在 DOS 提示符下打:

B>MASM AUTOSAVE,<CR>

B>LINK AUTOSAVE,<CR>

即生成可执行文件 AUTOSAVE.EXE。

### 四、使用

(一)在一张磁盘上建立文字编辑软件,并将 AUTOSAVE.EXE 拷入,插进 A 驱动器。打:

A>AUTOSAVE<CR>则程序自动在 A 盘根目录下生成一个名为 SAVE.DAT 的文件,在此以后的键盘输入都将记入这个文件里。现在,就可以进入文字编辑软件,放心地进行文字数据的录入了。

这时,即使突然停电,也不必担心,所录文字数据都已存在 SAVE.DAT 中了。只需读入这个数据文件,删去其中的一些多余的文字符号(这些多余的文字符号是由于您的输入错误引起的),即可继续往下进行文字数据的输入了。

(二)“黑盒子”功能。将 AUTOSAVE.EXE 拷入硬盘根目录下,在自动批文件 AUTOEXEC.BAT 中加入一条命令:

AUTOSAVE

即可。这时,每次启动后,在 SAVE.BAK 中即可查到上次启动后的键盘输入情况。

### 五、说明

(一)本程序的记录功能很强,对所有可显示的键盘输入,包括错误的输入,都能如实地记录下来。因此,当用来恢复文件时,对这些多余的文字符号应予删除。

(二)在“黑盒子”功能中,执行本程序的命令在自动批文件中的位置对记录的内容将有很大影响。如果

AUTOSAVE 是 AUTOEXEC. BAT 的第一条命令, 则 SAVE. DAT 中记录的将只是 ACSII 码, 或汉字的外码(输入码), 而不能记录汉字的输入。如果需要记录汉字, 则 AUTOSAVE 必须放在装载汉字系统的命令之后。例如, 执行了 WBZX(五笔字型装入模块)之后, 再执行 AUTOSAVE, 就可记录用五笔字型输入的汉字。

(三) 由于每次开机启动都将刷新 SAVE. DAT, 因此如果多次开机, 则最后只能看到上次开机后的键盘输入情况。为了全面记录一段时期内的键盘输入, 可以在 AUTOEXEC. BAT 中 AUTOSAVE 命令之前, 加上一条命令:

COPY SS+SAVE. DAT SS>NUL 这样, 在 SS 中就可查到一段时期以来的全部键盘输入。注意对 SS 文件进行定期清理, 否则其长度将迅速增加, 而占去很大的磁盘空间。

(四) 本程序在 PC 机上运行通过。

(五) 欢迎读者对本程序的不足之处, 及其功能改进, 提出宝贵意见。

```
cseg    segment para public
        assume cs:cseg, ds:cseg, es:cseg

stk     db    20h dup('stack')    ;堆栈区
count   dw    00h                  ;键盘计数
lim     dw    40h                  ;每次存盘字节数(可改)
int16   dd    00000h              ;原 INT 16H 中断向量
stk     dd    00000h              ;保存原栈段指针
file1   db    '\save.dat',0        ;数据文件名
file2   db    '\save.bak',0        ;数据文件名
int16n  proc far
        pushf                      ;保存标志
        or     ah,ah               ;非0号功能
        jnz    old                 ;转 old
        popf                       ;0号功能
        pushf
        call   dword ptr cs:int16;从键盘读字符
        pushf                      ;返回 ah=扫描码
        push   ax                  ;al=字符码
        push   bx                  ;保存寄存器及标志
        cli                      ;清中断
        mov    word ptr cs:stk,sp ;保存栈段指针
        mov    word ptr cs:stk+2,ss
        mov    bx,cs
        mov    ss,bx
        mov    sp,0FEh            ;修改栈段指针
        sti                      ;开中断
        push   cx
        push   dx
        push   ds                  ;保存寄存器
        push   cs
        pop    ds                  ;数据段指向代码段
        cmp    al,20h              ;是否为 ACSII 码?
        jnb    save               ;是,则转 save
        cmp    al,0dh              ;是否为0dh(回车)?
```

```
        jz     save               ;是,则转 save
        jmp    on                 ;否则,转 on
save:
        lea    bx,keep            ;bx 指向缓冲区首址
        add    bx,count           ;加上计数
        mov    [bx],al            ;字符码存入缓冲区
        inc    count              ;计数加1
        call   checkc             ;转子程序
        cmp    al,0dh             ;是否为“回车”
        jnz    on                 ;不是,转 on
        mov    al,0ah             ;是,
        jmp    save               ;则再存一个“换行”符
on:
        pop    ds
        pop    dx
        pop    cx                  ;恢复寄存器
        cli
        mov    bx,word ptr cs:stk d+2
        mov    ss,bx              ;恢复栈段指针
        mov    sp,word ptr cs:stk d
        sti
        pop    ax
        popf
        iret                      ;中断返回
old:
        popf
        jmp    dword ptr cs:int16 ;转原中断
checkc  proc near
        mov    cx,cs:lim          ;检验计数是否达到
        cmp    count,cx          ;每次存盘字节数
        jb     retc               ;未达到,则返回
        call   write              ;达到了,则转存盘程序
        retc:                    ;返回
        ret
checkc  endp
write   proc near
        push   ax
        lea    dx,[cs:file1]
        mov    ax,3d01h
        int    21h                ;打开 save.dat 供写
        jb     return             ;出错返回
        mov    bx,ax              ;文件句柄送 bx
        xor    cx,cx
        xor    dx,dx
        mov    ax,4202h
        int    21h                ;移动文件指针到文件尾
        jb     closef             ;出错,转 closef
        lea    dx,[cs:keep]
        mov    cx,cs:lim          ;字节数送 cx
        mov    ax,4000h
        int    21h                ;写文件
closef: mov    ax,3e00h
        int    21h                ;关闭文件
        mov    word ptr cs:count,0 ;计数清零
return: pop    ax
        ret                      ;返回
write   endp
keep    dw    20h dup(?)          ;键盘输入缓冲区
```

```
int16n endp
```

## 主程序

```
autosave proc far
start:  push cs
        pop  ds
        push cs
        pop  es
        lea  dx,file2
        mov  ax,4100h
        int  21h          ;删去 \save. bak
        lea  dx,file1
        lea  di,file2
        mov  ax,5600h      ;将 \save. dat 改名
        int  21h          ;为 \save. bak
        mov  ax,3c00h
        mov  cx,0
```

```
int  21h          ;重建 \save. dat
mov  ax,3516h
int  21h          ;取 INT 16H 中断向量
mov  word ptr int16,bx;保存 INT 16H 中断向量
mov  word ptr int16+2,es ;在 int 16 中
mov  ax,2516h
lea  dx,int16n      ;修改 INT 16H 中断向量
int  21h          ;使指向 int16n
lea  dx,keep
add  dx,lim
add  dx,0100h
int  27h          ;程序驻留退出
autosave endp
cseg  ends start
end
```

## PC 机软件加解密技术剖析(续)

华南理工大学计算机系软件专业88级 李文亮

加密和解密是对立的不可分割的两个方面。没有无所不摧的矛,也没有攻不破的盾,同样也不存在万能的加密和解密方法。一般说来,加密都是针对某些破译方法的,若知道了加密方法,解密相对就是很容易的了。因此,就需要编写反跟踪程序段,以求在最大可能的限度下,阻止破译者的跟踪分析。

反跟踪的方法是多种多样的,这里主要介绍一些常用的方法,在此基础之上,用户可以充分发挥自己的想象力,创造出更多更好的方法。

### 1. 反反汇编

通常程序在装入内存之后不再改变,因此用户可以用 DEBUG 等调试工具装入程序,然后利用反汇编命令得到程序清单。实用的加密系统则实现了对自身代码的逐段加密,在程序运行过程中逐块解密执行,并破坏已执行完的程序块。这样,任意时刻内存中都不存在完整的程序,只有当前程序块是可见的。对反跟踪程序的加密算法,一般要求算法简单易行,不要消耗太多的时间,而且要有较好的保密性。常用有以下几种方法:

(1)逻辑运算法。即将原代码与参量逐字节进行逻辑运算而形成密码。如采用逐字节异或(A XOR B=C 加密, C XOR B=A 解密)。如果参量也按一定的规律来生成,保密性更佳。

BASICA 文件的 P 方式存储就是根据这一原理来

实现加密的。下面来看 CCCC.COM 中的一个片断。

R

AX=160E BX=0195 CX=163E DX=0000 SP=E8DF

BP=0100 SI=01E5 DI=0000

DS=160E ES=160E SS=160E CS=160E IP=01DC

NV UP EI PL NZ AC PE NC

160E:01DC B94500 MOV CX,0045

-U 1DC 1E3

160E:01DC B94500 MOV CX,0045 ;这里数据段和代码段相同

160E:01DF 90 NOP

160E:01E0 000C XOR[SI],CL ;异或一个字节代码

160E:01E2 46 INC SI

160E:01E3 E2FB LOOP 01E0 ;循环执行

(2)位移法。这种方法通过将一个字节的循环信息左或右移若干位来实现。如 FBH 循环右移3位,就变成7FH。循环的位数也可以通过某一算法来确定,同时,PC 机上的循环移位可以通过一条机器指令来完成,既有较强的加密性能,实现也很方便。这种方法的另外一种形式是通过把原代码加或减去某一个数值来形成密码。

(3)交换法。通过把字或字节信息中的某几位进行互换,以达到加密的目的。

(4)错位法。即根据指定的代码对换表,或根据一定的规律把原码变换为另一代码。如 A→K, F→+, ...。这种方法的加密性能最强。但前者需要一块空间来存放对换表,而后者在规律被得知可轻易解密。

另外,还可以采用如 DES 等一些保密性更强的算法,但这样一来,解密过程就更为复杂,要以大量的时间为代价。由此可见,加密往往需要以时间和空间来换取加密性能。

单一地使用某种方法,是很容易被破解的。实用的加密系统都是巧妙地综合利用多种方法,实现分块

层次可变的加密,最后还往往需要利用密钥来参与解密运算,这样就大大地增强了解密的难度。

对付上述手段,解密时只能逐块跟踪,逐块解密;积少成多,最终一定可以看到程序的全貌。同时,还要记下一些重要参数,如当前块的首末地址,使用到的一些参数块地址等。

反汇编的另一种方法,就是利用 PC 机的一些特殊指令,实现指令的重叠;或者在程序中穿插一些无用代码,造成反汇编的失败。

对付这种方法,用 DEBUG 反汇编的效果最差,最好利用 SOURCER 中的代码类型 FRAGMENT 来分析这个片断。SOURCER 是一个高级反汇编工具,它可以直接从内存或文件生成原代码或列表文件,并有多种说明,效果比较好。

最后介绍一种“指令逆行”的方法,它几乎达到了无懈可击的地步,具有最强的保密性。这种方法是作者在彻底剖析过数个加密系统之后,发现并提出来的一种新型的加密方法,并已经在 IBM 机上具体成功实现。附录3提供了全部详细的程序清单。

通常 PC 上机程序都是由低地址向高地址逐条指令排列和执行的,用调试程序可以方便地进行反汇编和跟踪。但是如果程序代码不按这种方式排列,而是由高地址向低地址反向排列,运行时也按这种反向顺序进行,那么,程序就无法反汇编了。同时,跟踪程序总是按进行跟踪执行的(除了 TURBO DEBUGGER 可以在特定地情况下“逆向”执行指令外,但这种“逆向”还是要求指令是按由低地址向高地址顺序存储的),这样一来跟踪也就无法进行了。

借助于 PC 机的单步中断,可以很方便地实现以上构思。可以这样设想:当每一条指令执行完之后,将产生单步中断,利用该中断使一条指令复原,然后执行该指令;指令执行完后,又产生单步中断,根据两次中断的返回地址,即可算出该指令的长度,据此向低地址方向找到下一条指令的起始地址,然后再把指令复原,再执行,……。这样,程序的运行就好像是从高地址向低地址方向进行。

其实我们可以看到,指令代码是否逆向排列,程序是否逆向运行,这还不是关键。关键在于每条指令执行完后才产生下一条指令代码,每条指令都是在被执行时才恢复。这样一来,跟踪者就根本无法进行反汇编和跟踪,只能老老实实地一条一条指令地恢复。假如有数

百条指令,并且每条指令都被加密,那么,要把这么多的指令全部恢复过来,将是一件非常“痛苦”的事情。而加密者只需编写一小段程序,瞬间即刻完成加密,非常方便。根据这一设想,我们可以产生无数的方法,“指令逆行”只是其中的一种。

根据“指令逆行”的原理,逆程序段中一般不能含有循环和回向转移,因为在程序运行中已经将原来的代码破坏了;另外,中断调用时要作特殊处理,因为在中断处理程序中,指令是顺序排列执行的;同时,程序逆行过程中还要禁止硬件中断,防止在执行诸如 INT 08H 时钟中断等服务程序时逆行引起的错误。

PC 机上程序要进入单步状态要有两个条件:一是设置了单步中断服务程序,二是把标识寄存器(FLAG)的中断允许位置为1。FLAG 的第八位为单步中断位,此位为1时,每执行完一条机器指令之后,CPU 就会产生单步中断,进入单步中断服务程序。在 DOS 下单步中断只有“IRET”一条指令,DEBUG 等调试程序每次执行 T 或 G 时,都把单步中断01和断点中断03改向自己的中断服务程序,这导致被调试程序的单步和断点服务程序永远无法执行。这样一来,调试程序就无法对逆程序进行跟踪了。我们用修改中断向量来实现单步中断程序的设置,用 IRET 来实现把单步中断位置1。

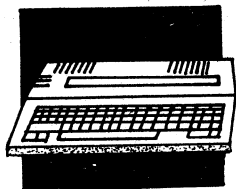
下面通过一个例子来说明该方法的实现。

由于8086系列芯片的指令最长为6个字节,所以在 INT 1中每次把指令代码前的5个字节移到指令代码后,就一定能组成一条完整的指令;同时,这样作还破坏了上一条指令。这里用代码0F 0H(指令“LOCK”)来作为逆程序的结束标志,每条指令都用逐字节异或0F 0H的方式加密。也可以采用别的方法。在遇到中断调用指令(指令代码以“0CDH”开头)的时候,必须改为顺序运行的方式,并在中断调用完成后再恢复为逆行方式。中断后的六条指令就是为此而特意安排的。要注意的是,这里中断调用结束后的返回地址(由 LEA AX,G1确定)必须作一些修改,因为逆行后指令地址已经发生了变化。而且,逆程序段的入口处那条指令执行前不会产生单步中断,指令应该顺序放置。

这段程序仅作演示用,在 TURBO ASM 2.01上编译连接成 COM 程序后,可正确执行,对其它的编译器,密码段的跳转地址可能要作修改。

(待续)





学习机之友

## 推荐一个扩展 BASIC 语句——拆字符串语句

北大分校数学系(100083) 王 凯

拆字符串语句是在 CEC 机或 APPLE 机上实现的一个扩展 BASIC 语句,其功能是将一个字符串拆成若干子串,它可以按任意指定的分隔符,灵活快速地将一个字符串分解,并把分解后所得的子串赋给指定的多个变量,因此具有广泛的用途。

### 一、从改善数据结构说起

以下程序可列表显示某班学生的选课成绩。

#### 【程序一】

```
10 HOME
20 PRINT TAB(3)“姓名”TAB(13)“课程 A”TAB(20)
   “课程 B”TAB(27)“课程 C”
30 READ N;FOR I=1 TO N
40 READ N$,F1$,F2$,F3$
50 PRINT TAB(3)N$ TAB(14)F1$ TAB(21)F2$
   TAB(28)F3$;NEXT
60 DATA 30
100 DATA 张兴华,98,,84
110 DATA 王 涛,,78,82
120 DATA 李 英,,,
```

在程序中小学生的姓名、各门选课成绩是由 DATA 语句提供的,数据必须按规定的格式输入:前后次序绝不可颠倒,缺项(未选某一门课)时必须连续地输入数据分隔符——逗号。这种严格的规定对于数据的输入、核查、修改、增删都带来了不便,特别是数据个数较多时更是如此。

一种理想的格式是:同一个学生的各门成绩前的分隔符能换成不同的字符以示区别,例如换成 A、B、C,以分别代表三门不同的课程;而且它们的次序可任意调换,缺项时也不必输入该项前的分隔符。按此设想,张兴华的成绩可表为 A98 C84 或 C84 A98,而李英三门课均未选,其成绩处可以什么都不写。

无疑这种格式有极大的改进,它被采用有赖于使用拆字符串语句,现将程序一做如下修改。

```
5 CH=37120;REM 拆字符串语句解释程序的入口地址
.....
40 READ H$;CALL CH EXP H$ ON “N$A F1$B F2$
   $CF3$”
.....
100 DATA 张兴华 C84 A98
110 DATA 王 涛 C82 B78
120 DATA 李 英
.....
```

修改后程序的功能完全没有变,但 DATA 语句中采用了理想的格式。

当循环变量 I=1 时,执行 40 句中的 READ 语句后,H\$ 的值为“张兴华 C84 A98”,再执行其后的 CALL 语句,实现了对拆字符串语句解释程序的调用,从而保留字 EXP 打头的拆字符串语句被执行,串变量 H\$ 被拆成:N\$=“张兴华”,F1\$=“98”,F3\$=“84”,而 F2\$ 为空串。

### 二、拆字符串语句的语法格式及基本功能

调用拆字符串语句的解释程序来执行该语句需采用以下复合语句:

(行号) CALL <入口地址> <拆字符串语句>

注意:CALL 语句与拆字符串语句之间只能有空格,不能把它们分为两个单独的语句。

拆字符串语句的语法格式为:

EXP<串变量> ON<串表达式>

这里 EXP 和 ON 是原 BASIC 保留字。EXP 之后的串变量是被拆串变量,ON 之后的串表达式经常是一个串常量或串变量,其值提供了拆字符串的依据,为简便起见,称此式的值为 ON 表,此表应符合以下规定:

<首串变量><字符1><串变量1>.....<字符 N><串变量 N>

其中的各串变量叫结果串变量,各字符叫分隔符,它们可以看作是其后的串变量的前缀,首串变量是不可缺的,而其后的分隔符和串变量必须成对。

结果串变量可以是简单变量,或下标变量,须注意的是:若其中的某一简单变量在执行此拆字符串语句前未曾赋值,那么在 ON 表中它绝不可排在任何下标变量之后!

除 CEC 机的汉字识别符(代码 \$7F)以外,分隔符可以是任意的 ASCII 字符,由于允许空格作分隔符,故 ON 表中不能有多余的空格。

当分隔符不只一个时,它们不必全相同,也不必全不同,下面说到“ON 表中某种分隔符的个数为 N”,指的就是该表中共有 N 个分隔符取的是同一个 ASCII 字符,此时表中可能还有别种分隔符。

拆字符串语句的基本功能有如下述。

1. 完全分解的情况,此时 ON 表中每一种分隔符的个数都不少于被拆串中该种字符的个数。

首先,解释程序依 ON 表提供的分隔符,把被拆串拆成若干子串(可能有的是空串),除首子串外,每个子串前都有一个 ON 表内的分隔符,可把它看作是此子串的前缀,每个子串内都不含任何分隔符,此时被拆串得到了完全的分解。

第二步,把首子串赋给首结果串变量,把其余每

子串(不包括前缀)按下述规定赋给与它有相同前缀的结果串变量:对同一种前缀而言,依由前至后的次序,把第K个子串(空串也算数)赋给第K个结果串变量。

若ON表中某种分隔符的个数N多于被拆串中所含的该种字符的个数M,则后面的N-M个结果串变量变为多余的,它们均得到空串值,特别地,若被拆串为空串,则每个结果串变量均得到空串值。

执行下面的程序,可将含有加、减运算的算式拆开。

#### 【程序二】

```
5 CH=37120
10 HOME:O$="B$+Z1$+Z2$-F1$-F2$=J$
20 READ A$:IF A$="00" THEN END
30 CALL CH EXP A$ ON O$
40 PRINT "B$:"B$,"Z1$:"Z1$,"Z2$:"Z2$,"F1$:"F1$,"F2$:"F2$,"J$:"J$
90 GOTO 20
100 DATA 7+8-6-4+5=10
110 DATA 7+6-3=10
120 DATA 7-3+6=10
190 DATA 00
```

执行以上程序,三个算式均被完全分解,分解第一个算式将显示出:

```
B$ : 7, Z1$ : 8, Z2$ : 5, F1$ : 6, F2$ : 4, J$ : 10
分解第二、三两个算式后,显示出的结果均为:
B$ : 7, Z1$ : 6, Z2$ : 1, F1$ : 3, F2$ : 1, J$ : 10
其中 Z2$、F2$ 的值为空串。
```

2. 不完全分解的情况,ON表中某分隔符的个数N少于被拆串中该种字符的个数M。

在此情况下,在被拆串所含的M个该种字符中,多余的后M-N个将失去分隔符的作用,因此在拆得某些子串内还含有该种字符,被拆串得不到完全的分解。

例如在程序二中,增加以下两行:

```
130 DATA 8-7+4+2-5+3=5
140 DATA 10-7-2+4-3+4+2=8
```

执行程序二后,以上二算式均得不到完全分解。

在130句的算式中第三个加号失去了分隔符的作用,结果F2\$的值为:"5+3",在140句的算式中第三个加号和第三个减号均失去了分隔符的作用,结果Z1\$、Z2\$的值分别为:"4-3","4+2"。

不完全分解的情况似乎是完全不能容忍的,但实际上并非如此。假如我们在编一个题库检索程序时,允许用户通过键入题号的方式来检索,此时可在程序中用INPUT语句接收用户的键入。为方便用户,最理想的是规定用户的键入采用如下的格式。如检索第5至8题、第10题、第21题就键入:5-8;10;21及回车键,此时在INPUT语句中只需用一个串变量,例如A\$,来接收用户的键入,然后用拆字符串语句分两步(先按分号再按逗号)分解串变量A\$,就可得到用户要检索的全部题号。

显然在做第一步分解时会产生这样的问题:如果ON表中的分隔符过多,常常会降低程序的效率,反之,不免出现不完全分解的情况。

其实这里有一个好的办法可解决困难,ON表中只需一个分隔符,可以多次执行拆字符串语句,逐步将A\$完全分解,读者不难看懂下面的程序。

#### 【程序三】

```
.....
400 FOR I=1 TO N:N%(I)=0:NEXT
410 INPUT "请按规定格式键入要检索的题号:";A$
420 IF A$="" THEN 470
430 CALL CH EXP A$ ON "T$;A$"
440 CALL CH EXP T$ ON "T$-U$";IF U$="" THEN U$=T$
450 T$=VAL(T$);U$=VAL(U$)
460 FOR I=T TO U:N%(I)=1:NEXT;GOTO 420
470 .....
```

在以上程序中下标变量N%(I)用来记录用户的选题情况,为了简明,程序中没有加入具有处理用户误操作一类功能的程序行。

#### 三、拆字符串语句的附加功能

在程序二中,分解算式7+6-3=10与算式7-3+6=10所得的结果相同,有时我们可能对此感到不满,希望知道是先加后减还是相反。其实拆字符串语句的附加功能完全可以满足这样的愿望。

我们约定凡下面提到的序号均是由左至右,从0起数而得的。假设分解被拆串后,ON表中第K个串变量所得的子串在所有子串中的序号为Z(K),此子串的首字符在被拆串中的序号为S(K),则

```
Z(K)=PEEK(512+3*K)-128
S(K)=PEEK(513+3*K)
```

应说明的是:因要查看的单元位于键缓区,故只有在执行拆字符串语句后,还没有新的键入前查看,查到的值才有效,再有,如果某一结果串变量是多余的,则求得的Z(K)、S(K)的对应值自然失去原来的意义,此时Z(K)必<0,而S(K)的值是随机的。

在程序二中增加以下程序行:

```
50 PRINT "Z(K):";FOR K=0 TO 5:PRINT PEEK(512+3*K)-128";NEXT;PRINT
60 PRINT "S(K):";FOR K=0 TO 5:PRINT PEEK(513+3*K)";NEXT;PRINT
```

对以上所述的二算式,将分别列出Z(K)、S(K)的值:

```
... ..
Z(K):0 1 -85 2 -83 3
S(K):0 2 211 4 225 6
... ..
Z(K):0 2 -85 1 -83 3
S(K):0 4 211 2 225 6
... ..
```

因算式中加、减号各只有一个,第二、四两个结果串变量Z2\$、F2\$是多余的,故Z(2)、Z(4)为负值,Z(2)、S(2)、Z(4)、S(4)均无意义。

利用拆字串语句的基本功能和附加功能,对一个字串进行一次或多次分解,常可较快地得到此字串内容的完整准确的分析。以下程序可用来检查一个字串中是否有大写元音字符:A、O、U、I、E,如果有则将其中的第一个这种字符及其位置序号列印出来。

#### 【程序四】

```
5 CH=37120:F$(1)="A":F$(2)="O":F$(3)="U":F$(4)="I":F$(5)="E"
10 HOME:INPUT A$:I=1
20 CALL CH EXP A$ ON "B$"+F$(I)+"C$"
30 IF A$ <> B$ THEN PRINT "第一个大写元音字符为:"F$(I),"其位置序号为:"PEEK(516):END
40 I=I+1:IF I<6 THEN 20
50 PRINT "不含任何大写元音字符"
```

有些读者可能会想到,用 MID\$( ) 函数也可以实现同样的功能,但那样做,程序中必须要增加一重循环,执行的速度肯定也要慢得多。

#### 四、利用拆字串语句可大量节省内存占用

假设我们要编一个人事档案检索程序,档案内容保存在磁盘上,关于每一个人的信息分属姓名、性别、

……、单位、职务、……等 N 项,其中可能含可省缺项,为了能灵活快速地检索,通常在检索前要把一个文件完整地调入内存,现在让我们来看看,在此拆字串语句能带来什么好处。

首先,拆字串语句可改善数据格式,给建文件带来方便,第二,此语句还可用来节省内存的占用。

如果不用拆字串语句,N 项信息得作为 N 个字段存于磁盘,调入内存时得占用 N 个下标变量,如果使用此语句,就可根据每一项的长短变化及是否可省缺,将 N 项信息组成较少的 M 个字段存于磁盘,调入内存时只占用 M 个下标变量。

一个串下标变量在变量表中要占三个单元,一个人的信息少占 N-M 个下标变量。假设 N-M=8,人数为 50,便可节省近千个单元,对 CEC 机或 APPLE 机来说,这不能说是个小数目。

#### 五、拆字串语句的解释程序

解释程序在内存中是可浮动的,不应忘记的是:将其调入内存后,常需及时利用 HIMEM 语句来保护它。

```
9100-A9DD 20 C0DE 20 E3DF
9108- 20 60DDA0 00 B1 83 48
9110-C8 B1 83 48 C8 B1 83 48
9118-A9 B4 20 C0DE 20 7BDD
9120- 20 FDE5 A8 F0 5E A5 5E
9128-D0 02 C6 5F C6 5E A9 00
9130- 99 01 02 B1 5E 99 00 02
9138- 88 D0 F8 A5 B8 48 A5 B9
9140- 48 84 B8 A9 02 85 B9 A2
9148- 00 A9 80 9D 00 02 8A 48
9150- 20 B1 00 20 E3DF 20 6C
```

```
9158-DDA0 00 98 91 83 68 AA
9160-E8 A5 83 9D 00 02 E8 A5
9168- 84 9D 00 02 E8 A5 B8 D0
9170- 02 C6 B9 C6 B8 B1 B8 C9
9178- 20 F0 D0 20 B1 00 F0 07
9180-C9 7F D0 C7 4C C9 DE 68
9188- 85 B9 68 85 B8 68 85 9F
9190- 68 85 9E 68 F0 68 85 9D
9198-A9 80 85 A7 86 A1 84 A0
91A0- 84 A6 B1 9E C9 7F D0 04
91A8-C8 C8 D0 0E A2 00 DD 00
```

```
91B0- 02 F0 0E E8 E8 E8 E4 A1
91B8- 90 F4 C8 C4 9D 90 E3 B0
91C0- 0A E6 A7 A5 A7 9D 00 02
91C8- 20 84 A6 84 A5 8A A6 A0
91D0- 85 A0 BD 01 02 85 A2 BD
91D8- 02 02 85 A3 98 A0 00 E5
91E0-A6 91 A2 C8 A5 A6 9D 01
91E8- 02 18 65 9E 91 A2 C8 A5
91F0- 9F 69 00 91 A2 A4 A5 C8
91F8-C4 9D 90 A4 F0 CB 60
```

## 也谈内存信息的打印

北京航空航天大学机械厂(100083) 张亭

本刊91年第3期《超级8~40单元打印》一文中介绍的程序弥补了监控打印内存信息功能的不足,但该程序仍然存在一些缺点:只适用于80列打印纸,若用132列打印纸还是存在相当大的浪费;打印行长不能少于8个单元,因此无法采用边角空白的废打印纸;存放在\$2000~\$20A6并不能避免与其他机器语言相冲突;不能连续打印多份;程序功能过剩,有嫌冗长,达到了167个字节。

本人以46个字节的简短程序解决了上述问题:可在一行中打印1~256之中任意个单元;可浮动,移到内存的任意空闲地址都能照常运行;运用简单的技巧就可连续多份打印。

使用之前,将首地址存\$06、\$07、末地址存\$08、\$09,程序首地址加上\$0A的单元(本例中为\$30A)存放一行打印的字节数,0表示256个(本例中为24,适用于80列打印纸)。

很多缺乏经验的人一听说打印就先用 PR # 立即命令接通打印机,结果把以后的操作步骤全部记录在打印纸上。本人主张不预先联机的打印,具体办法如下:

| 键入本程序,并正确置好首尾地址、一行字节数之后 |                                                             |
|-------------------------|-------------------------------------------------------------|
| 监控程序中                   | * 1P=300G                                                   |
| BASIC 状态有 DOS           | ]PR #1;FOR I=1 TO 3:CALL 768:?:?:NEXT                       |
| BASIC 状态无 DOS           | ]PR #1;FOR I=1 TO 3:CALL 768:?:?:NEXT:PR #0                 |
| 备 注                     | 1. 打印机假设在1号槽<br>2. P=表示 CTRL 与 P 同时按下<br>3. 假设共需打印3份,每份间隔2行 |

当该程序移到其他地址运行时,表中的300G和CALL768就都要作相应的调整。

0300— A0 03 B6 06 96 3C 88 10—F9 A9 18 85 FF 20 92 FD

0310— A9 A0 20 ED FD B1 3C 20—DA FD 20 BA FC B0 0E

C6

0320— FF F0 E6 A5 3C 29 07 D0—E7 A9 AD D0 E5 60

## 增强中华学习机的音响功能

广东佛山永安路154号(528000) 杨云霄

中华机的发声电路只具备微机发声功能的最起码条件,只能奏出单调、无伴奏的单声部乐音。这与现今游戏机相比逊色多了,从而极大地限制了教学、游戏软件的音乐渲染力。虽然已有与它配套的音响合成部件出售,但价格昂贵、使用并不方便等,使音乐卡没能被大多数用户接受。

笔者从软件上着手,采用与众不同的发音原理编写这个程序。它大大地改善了主机的音响效果,使之可奏出双声部,各声道独立,并且可控制声道开或关、演奏速度可变、可奏出七个八度音(一般程序只能奏三或四个八度音)。在不改变原有乐音数据前提下,采用改变各声道音调控制单元的值,还可使同一首歌曲用几十种音调奏出,甚至和声。它可作为音乐程序中的子程序调用(用CALL24576或JSR\$6000)。

程序说明:音乐数据按音阶①音长①音阶②音长②(①、②各表一声道)存放。当乐曲为单声部时则令音阶②为音阶①高一个八度,对于乐曲为双声部但音长②与音长①不相等时则把较大值的那一个分为n个的出的个数与较小值的那个声道的下一个乐音按发音格式组合,其它如此类推。程序中\$6029单元控制演奏速度。\$602F、\$6030单元各控制一个声道的声调,它们的值均在\$00—\$FF之间,其对应单元的值越大则速度越慢、演奏音调越高。特别是当\$602F、\$6030单元的值\$00时则关闭对应声道。音乐数据的存放是从\$610F始,以两个\$00值为结束。本程序设有录音输出,用于扩大音量,使音乐更动听。

乐谱数据表只列出C大调(即\$602F、\$6030两单元值为\$CC,其二进制代码11001100,“1”表示触发扬声器,“0”不触发)的四个八度。可通过改变这两个单元值来提高或降低,以获得七个八度。所有的拍子均分成较短的来演奏,使两声道和谐地交融在一起。

笔者打算近期内介绍其音乐数据生成的快速程序。同时,已编好几十首世界名曲,免费提供给读者。

乐谱数据表

|          |          |          |          |            |
|----------|----------|----------|----------|------------|
| 1        | 1 \$ B5  | 1 \$ 56  | 1 \$ 26  | 1/16拍 \$ 3 |
| #1       | #1 \$ AB | #1 \$ 50 | #1 \$ 23 | 1/8拍 \$ 7  |
| 2        | 2 \$ A1  | 2 \$ 4B  | 2 \$ 21  | 1/4拍 \$ 10 |
| b3       | b3 \$ 97 | b3 \$ 47 | b3 \$ 1E | 2拍 \$ 22   |
| 3        | 3 \$ 8E  | 3 \$ 42  | 3 \$ 1C  | 4拍 \$ 52   |
| 4        | 4 \$ 85  | 4 \$ 3D  | 4 \$ 1A  |            |
| #4       | #4 \$ 7D | #4 \$ 3A | #4 \$ 18 |            |
| 5 \$ F5  | 5 \$ 76  | 5 \$ 36  | 5 \$ 16  |            |
| #5 \$ E7 | #5 \$ 6F | #5 \$ 32 | #5 \$ 14 |            |
| 6 \$ DA  | 6 \$ 68  | 6 \$ 2F  | 6 \$ 13  |            |
| 7 \$ CD  | 7 \$ 61  | 7 \$ 2C  | 7 \$ 11  |            |
| 7 \$ C1  | 7 \$ 5B  | 7 \$ 29  | 7 \$ 0F  |            |

注:对应为空格者表示此调溢出

|        |    |    |    |    |    |    |    |    |
|--------|----|----|----|----|----|----|----|----|
| \$6000 | AE | 2F | 60 | 8E | 79 | 60 | A9 | 01 |
| \$6008 | 20 | 35 | 60 | 20 | 4E | 60 | F0 | 08 |
| \$6010 | AD | 30 | 60 | CD | 79 | 60 | D0 | 03 |
| \$6018 | AD | 2F | 60 | 8D | 79 | 60 | AD | 00 |
| \$6020 | C0 | 10 | E8 | 8D | 10 | C0 | 60 | 0F |
| \$6028 | 61 | B9 | 00 | 00 | 00 | 00 | 00 | CC |
| \$6030 | CC | 00 | 00 | 00 | 00 | AE | 29 | 60 |
| \$6038 | 8E | 2A | 60 | 0A | AA | BD | 25 | 60 |
| \$6040 | 85 | 00 | 8D | 33 | 60 | BD | 26 | 60 |
| \$6048 | 85 | 01 | 8D | 34 | 60 | 60 | A4 | 10 |
| \$6050 | D0 | 10 | A0 | 01 | B1 | 00 | D0 | 0D |
| \$6058 | AD | 33 | 60 | 85 | 00 | AD | 34 | 60 |
| \$6060 | 85 | 01 | A9 | 00 | 60 | 0A | 0A | AA |
| \$6068 | AD | 2A | 60 | 20 | F0 | 60 | 8E | 2D |
| \$6070 | 60 | 8D | 2C | 60 | 88 | B1 | 00 | A8 |
| \$6078 | A9 | CC | 20 | 8B | 60 | A5 | 00 | 18 |
| \$6080 | 69 | 02 | 85 | 00 | A5 | 01 | 69 | 00 |
| \$6088 | 85 | 01 | 60 | E0 | 00 | F0 | 4B | C0 |
| \$6090 | 00 | D0 | 04 | A9 | 00 | A0 | 01 | 8D |
| \$6098 | 2B | 60 | 8C | 2E | 60 | 8E | 2D | 60 |
| \$60A0 | A9 | 00 | 8D | 2C | 60 | A2 | 08 | 20 |
| \$60A8 | DB | 60 | AC | 2E | 60 | 88 | D0 | FD |
| \$60B0 | CA | D0 | F4 | AD | 2C | 60 | 38 | ED |
| \$60B8 | 2E | 60 | 8D | 2C | 60 | AD | 2D | 60 |
| \$60C0 | E9 | 00 | 8D | 2D | 60 | 90 | 13 | AD |
| \$60C8 | 2C | 60 | 38 | E9 | 08 | 8D | 2C | 60 |
| \$60D0 | AD | 2D | 60 | E9 | 00 | 8D | 2D | 60 |
| \$60D8 | B0 | CB | 60 | 0E | 2B | 60 | 90 | 0A |
| \$60E0 | EE | 2B | 60 | 2C | 30 | C0 | 2C | 20 |
| \$60E8 | C0 | 60 | 48 | 68 | EA | EA | EA | 60 |
| \$60F0 | 8D | 31 | 60 | 8E | 32 | 60 | A9 | 00 |
| \$60F8 | A2 | 08 | 4E | 31 | 60 | 90 | 04 | 18 |
| \$6100 | 6D | 32 | 60 | 6A | 6E | 31 | 60 | CA |
| \$6108 | D0 | F3 | AA | AD | 31 | 60 | 60 |    |



# 计算各类债券储蓄收益利息 BASIC 程序

南京物资学校(210003) 陶文庆

随着改革开放发展经济的需要,城乡人民储蓄不断增长,国内证券市场正进一步开放、发展,购买国库券、企业债券的个人和单位越来越多。在债券发行时以面额价购债券的收益以及各类储蓄的利息,需要计算以予知或复核其效益;而中途以高于面额的款额从转让机构买入二手债券,其收益和收益率则是购买者更为关心的。怎样快速、准确地算出有关购、储的收益,以决定是购,是储?购多少,储多少?

这里提供了一个适用于各类储蓄、发行债券、二手债券收益利息计算的 BASIC 程序。只需要按微机提示键入相应内容的数据,电脑就会“动脑筋”帮助你,1、计算各类储蓄的利息、本息合计;2、计算购买各种发行期债券的到期利息和本息合计;3、计算中途买入二手债券的利息、本息合计及所购二手债券的后期(到期)收益率。

操作中,电脑要求键入“年利率%(.#. #.);月息%(-#. #.)”,这时如果是给月息,请在数据前加负号,如月息六厘六(6.6‰),应键入“-6.6”,以便识别,给年利率,则输入百分号前的数据即可;如年利率为10%,应键入“10”,年利率为9.5%,应键入“9.5”。为方便计算,本程序以月为单位计天数,一年按360天,一月按30天计。程序已在紫金 I 及 LASER310上运行验证,其他机型也可移植。

```
10 INPUT "中途买入请键入8,发行期买债券或储蓄请键入9",B
20 INPUT "年利率%(.#. #.);月息%(-#. #.)",L;
  IF L < 0 THEN L = -(L/1000);GOTO 30
25 L = L/100/12
30 INPUT "发行(存入)年,月",F1,F2
40 INPUT "到期年,月",D1,D2
50 INPUT "买(存)入(债券为面额)多少元?",N
60 IF B=9 THEN 200
70 INPUT "每100元买入价(元)",R;R=R/100
80 INPUT "买入年,月",Z1,Z2
90 GOSUB 260
100 RPINT "到期所购债券本息合计",N+(N*L*Y);"元"
110 PRINT "其中收益(利息)为",N+(N*L*Y)-(N*R);"元"
120 PRINT "您买这些债券用了",N*R;"元"
130 PRINT "到期收益率是:";
140 IF Z2=D2 THEN T=(D1-Z1)*12*30
150 IF Z1=D1 THEN T=(D2-Z2)*30
160 IF Z2>D2 THEN T=((D1-Z1-1)*12+D2+12-Z2)*30
```

```
170 IF D2>Z2 THEN T=((D1-Z1)*12+D2-Z2)*30
180 PRINT (N+(N*L*Y)-(N*R))/(N*R)/T*360
  *100;"%"
190 RUN
200 GOSUB 260
210 PRINT "到期利息是:",N*L*Y;"元"
220 PRINT "本息合计=",N+(N*L*Y);"元"
240 RUN
260 IF F2=D2 THEN Y=(D1-F1)*12
270 IF F1=D1 THEN Y=D2-F2
280 IF F2>D2 THEN Y=(D1-F1-1)*12+D2+12-F2
290 IF D2>F2 THEN Y=(D1-F1)*12+D2-F2
299 RETURN
```

## 哥德巴赫猜想 BASIC 程序的改进

浙江金华县傅村镇校(321037) 卢良红

本刊1991年第3、第10期发表了陈君佐同志的关于验证 Goldbach 猜想的 BASIC 程序,文后所附的大量结论,在我校师生中引起了强烈反响。笔者连忙将它送入中华学习机中运行,却意外的发现其运行速度实在太慢了,特别是奇数猜想,在探求1991的3285组解时,竟花费了近3个小时。仔细分析原程序,发现在每探求一解时,都要从头计算各质素数,耗费了不少无效循环运算,致使运行速度慢、且不易连续探求。笔者经多方编程试验,终于找到了解决的方法:首先采用素数的快速求解法,把所需验证的N以内所有质素数都存在数组A%(4000)之中,然后采用数据搜索法求解。这样大大减少了无效循环运算,节省了占机时间,提高了程序运行效率。在验证1991年的3285组解时,仅耗时12分10秒。在偶数猜想中,将数从6验证到2000实际耗时仅53分钟,速度比原程序提高了十余倍。

```
10 INPUT N;DIM A%(4000);A%(1)=3;K=1
20 FOR I=6 TO N+1 STEP 6:FOR J=I-1 TO I+1 STEP
  2;E=SQR(J);FOR L=1 TO K:IF (J/A%(L))=INT
    (J/A%(L)) THEN 60
30 IF A%(L)>E THEN 50
40 NEXT L
```

```

50 K=K+1:A%(K)=J
60 NEXT J,I
70 PRINT "1,偶数猜想";PRINT"2,奇数猜想";PRINT"3,
  END"
80 PRINT "请选择1,2,3":GET  A:ON A GOTO 90,160,
  260
90 L=2,FOR I=6 TO N STEP 2:P=0
100 IF A%(L)<I-3 THEN L=L+1,GOTO 100
110 E=L,FOR J=1 TO K:IF A%(J)>I/2 THEN 150
120 F=A%(J)+A%(E):IF F>I THEN E=E-1,GOTO
  120
130 IF F=I THEN P=P+1,PRINT P;"I"="A%(J)+"
  A%(E)
140 NEXT J
150 NEXT I,GOTO 70
160 L=2,FOR M=9 TO N STEP 2:A=0,C=1
170 IF A%(L)<M-6 THEN L=L+1,GOTO 170
180 D=L,FOR I=1 TO K:B=0,IF A%(I)>M/3 THEN
  PRINT "END",GOTO 250
190 E=D,FOR J=I TO K:IF A%(J)>(M-A%(I))/
  2 THEN 240
200 F=M-A%(I)-A%(J)
210 IF F<A%(E) THEN E=E-1,GOTO 210
220 IF F=A%(E) THEN A=A+1,B=B+1,PRINT A "
  C " "B" "M"="A%(I)+"A%(J)+"A%(E):IF J=I
  AND B=1 THEN D=E
230 NEXT J
240 C=C+1,NEXT I
250 NEXT M,GOTO 70

```

## 数组的动态删除

成都商业学校(610091) 冯端品

在PC机的BASIC语言中有一条ERASE语句,可以把不再需要的数组删除,以后又可以重新定义。这对于使用大型数组的程序节省内存空间是非常有用的。但是苹果机、中华学习机等却没有相应的语句,这

对于内存容量本就很小的八位机来说,实在是一个很大的遗憾。

我分析了苹果机、中华学习机上变量在内存中的存储结构后,发现只要对数组变量区的数据作适当的调整,同时修改相应的指针,就可以起到删除数组的作用。于是我用6502机器语言编了一段程序,解决了这个问题。

首先运行程序(\*300G或]CALL768),然后在你的程序中要删除某个数组时,先使用该数组一次(例如给数组中任意一个元素赋值),后面跟冒号和&命令,即可删除该数组。

后附的BASIC程序是删除数组的一个例子,其中30句就是删除数B,40句重新定义了数组B,且下标上界比原来的大,程序照样能顺利运行。

可以删除任何类型的数组,不管是实型数组还是整型数组,不管是数值型还是字符型,也不管它们是几维的,都可以删除。但一次只能删一个数组。

机器语言程序的地址是浮动的,可以移往任何区域,只是原来\$301、\$306单元的数据要作相应的调整,使它们指向新的入口地址。

```

300— A9 0B 8D F6 03 A9 03 8D
308— F7 03 60 A5 6B 85 42 A5
310— 6C 85 43 A5 6D 85 3E A5
318— 6E 85 3F A0 00 B9 81 00
320— D1 42 D0 08 C8 B9 81 00
328— D1 42 F0 21 A0 02 18 B1
330— 42 65 42 48 C8 B1 42 65
338— 43 85 43 68 85 42 A5 43
340— C5 3F 90 D7 D0 07 A5 3E
348— C5 42 B0 CF 60 A0 02 38
350— A5 6D F1 42 85 6D C8 A5
358— 6E F1 42 85 6E A0 02 18
360— A5 42 71 42 85 3C C8 A5
368— 43 71 42 85 3D A0 00 20
370— 2C FE 60
10 DIM A$(1,2),B(3),C%(2,1)
20 A$(1,0)="ABC":B(1)=1,C%(0,1)=2,PRINT A
  $(1,0),B(1),C%(0,1):PRINT
30 B(0)=0:&
40 DIM B(5),B(5)=7:PRINT A$(1,0),B(1),C%(0,1),
  B(5)
50 END
RUN
ABC      1      2
ABC      0      2
7

```

# 第一讲 概述

北京西四北三条10号 FORTH 应用研究会(100034) 丁志伟

应部分读者要求,从本期开始,我们将连续刊载丁志伟先生撰写的 FORTH 语言介绍,有关 FORTH 语言的特点及学习意义,在这篇概述中已有很好的说明,不再赘述。

前一时期,笔者在杂志上发表过几篇介绍 FORTH 语言的文章。这种语言的独到之处,引起了许多读者的兴趣,因此陆续收到不少来信,询问有关信息、寻找资料及软件,想进一步了解这个语言。FORTH 有很多特点,与 BASIC、PASCAL 和 C 语言差别较大,相比之下其应用不如这些语言那么广泛,资料也较少。据笔者知,有关书籍在国内出过十几种,大多已无法得到。港台美日资料也有一些。各种计算机刊物上有关 FORTH 的文章虽时有所见,但较分散,找起来并不容易。

软件方面,由于 FORTH 语言实现和改造比较方便,因此版本较多,估计国内至少有二十多种,较为流行者,APPLE 机上的,有 FIG FORTH、FORTH79、FORTH I 等;PC 机上,有 PC/FORTH2.0 版和 3.2 版、F-PC3.5 等。这个讲座就是以 APPLE 机上的 FORTH I D1.0 和 PC 机上的 PC/FORTH2.0 为基础,介绍 FORTH 语言。

本讲座是为了帮助广大想学习 FORTH 而又缺乏资料的读者而办的,考虑到《电子与电脑》读者有众多业余爱好者,因此将介绍得较为详细具体。

如果需要资料及软件,请与北京西四北三条10号,中国软件行业协会 FORTH 应用研究会秘书处联系(邮编100034),也可以打电话:655661。

FORTH 是一种计算机高级语言。

计算机语言出现已近半个世纪,发展到今天,已经出现过上千种高级语言。其中人们普遍使用的有几十种。FORTH 也在这几十种之中。每一种高级语言都有其特点。比如 BASIC 和 LOGO 是为初学者设计的,FORTRAN 是科学计算语言,系统设计常用 PASCAL 和 C 语言,COBOL 多用于事务处理,LISP 和 Prolog 是人工智能语言,Post Script 是一种类似 FORTH 的语言,多用于排版和高级打印机,等等。

FORTH 是什么样的语言?为什么叫做 FORTH?它有什么特点?已经有那么多高级语言,为什么还要学习使用它呢?这些问题,正是这个讲座所要解答的。

### 名称及由来

FORTH 语言是由美国计算机专家 Charles

Moore 在60年代于美国佛吉尼亚无线电天文台工作时发明的。当时将其用于控制巨大的射电天文望远镜。他在 FORTH 名著《STARTING FORTH》(中译本《FORTH 入门》)的前言中是这么讲的:

若干年前,我是把 FORTH 做为程序员和计算机之间的操作界面而开发的。

传统的计算机语言无法满足我的各种要求。诸如功能、灵活性和简捷等等。为了使这个语言全面包含高水平程序员所需要的各种性能,我在设计 FORTH 时没有遵循其他语言中的常规做法。这些性能中最重要的一点就是其扩充能力,即凡是所需要的功能都能被加入到原有语言系统之中。

当我第一次把各种设想组合成一个程序实体,在当时的“第三代计算机”IBM1130上运行,其效果如此之好,以至于我认为它是“第四代计算机语言”。最初想把它命名为 FOURTH(第四代),但由于 IBM1130 机只能接受5个字符的标识符,于是将 FOURTH 改为 FORTH,这就是其名称的由来。

正因为如此,人们又常把这种语言称为第四代计算机高级语言。顺便说一句,FORTH 这个词,在英文中是向前的意思,这种语言确实有一种向前的精神。它虽不象 BASIC、C 等语言那样风行,却也吸引了众多爱好者,一直在稳步发展。关于计算机语言的第几“代”如何划分,是有不同看法的,这里不做讨论。不过 FORTH 的确具有一系列特点,足以使它与其他高级语言相区别,也正是由于这些特点,一些人感到难以接受,另一些人又会发现,某些用其他语言不易解决的问题,用 FORTH 来解决却很方便,或者在其他语言无法使用的场合,这个语言却能很好地工作。

### FORTH 语言的特点

FORTH 语言在概念、结构、程序表现形式及使用操作各方面都与大多数高级语言有不少差别。它具有一系列特点。如:扩充能力极强、词典结构、堆栈及相应的逆波兰表示法、完全结构化、非常高的时空效率、独特的解释/编译方法、穿线代码、溶汇于高级语言的结

构化汇编和可以方便地反编译,等等。这些特点,又是互相关联的。由于这些特点的存在,实际了解这个语言,对于喜欢接受新事物的人,能发现这是一片新天地;对于只了解传统高级语言的人,也会有耳目一新的感觉,能加深对计算机语言本质的认识。

在众多特点中,最重要的是对系统的可扩充(或者说可扩展)能力。限于篇幅,这里无法全面介绍各种特点,下面将重点介绍系统扩充,同时介绍其他与之关联的特点。

为了使问题更清楚,先来看看不具备扩充能力而大家又较为熟悉的 BASIC 语言在这方面的情况。比如,某些 BASIC 只能使用10进制数,而在与机器深层系统打交道时,使用16进制会更方便。这时会需要编一段10进制~16进制转换子程序。这样的子程序较为通用,可以用于别处,编其他程序时可以将其搬过来加以使用。随着时间及经验增长,每个编程者都会积累一批这样的通用子程序。它们相当于制造业中的通用部件,利用它们可以减少编程工作量。

可是事情并不完美。使用子程序,总要把它和主程序一同使用,程序接口常常是容易发生问题的地方。变量的用法,参数如何传递,都有可能出错。虽说编写工作量减少了,但仍要精心对待。同时,由于子程序要与主程序一同使用,程序规模不但不会因为编程量减少而缩小,反而更加容易膨胀。随着程序规模增加,处理上也会越来越复杂。这样的语言,程序和系统是截然分开的,系统本身不会随着编程者的经验增加而变化。好象旅行,每次只能从一个起点出发,越是远行,就非得背上更大的包袱。

很多人都看到了这点。自然会产生一种愿望:能否有这样的一种方法,所需的功能,只要编一次程序,系统就会记住,以后用到时只要直接引用就行了,即系统具有学习能力。遗憾的是,大多数语言做不到这点,它们的功能是固定的,不对用户开放。其实想开放也难以实现,这些语言内部结构太复杂,一般人难以理解。APPLE II 机问世以来,不知有多少人对 BASIC 解释程序做过剖析,一部分人还做过改进工作,但成绩并不大,一般不过是打几块补丁而已。现在人们普遍使用的仍然是厂家提供的固化 BASIC。随着软件技术的发展,现在虽然已经出现了 PC 机的 Turbo C 这样的系统,操作上很方便,可以使用众多函数库。但问题的实质并没改变:系统本身的功能都是由厂家提供的,用户只能被动接受,想改进还是不太可能。

现在来看看 FORTH。

首先举一个简单的实际例子。本文写作期间,笔者正在使用 PC/FORTH 2.0 编程,这是国内 PC 机上最普遍的版本。它能利用 PC 机的彩色显示方式,但平常却并没有使用彩色。为了改进显示效果,笔者定义了一组色彩功能模块。这种模块在 FORTH 语言中被称为 WORD,国内翻译成“单词”,也叫“词”或“字”,这个版本提供的几个与色彩有关的词中,有两个很重要。一个是 FOREGROUND,用于改变前景颜色,它决定显示

字符的颜色;一个是 BACKGROUND,用于改变背景。想改变以后显示字符的颜色,先给出一个颜色编号,再使用 FOREGROUND。比如红色编号是4,要用红色,就可在键盘上打入

4 FOREGROUND

以后显出的字符就是红色了。

利用这个 FOREGROUND,就可以定义出一组颜色词。程序是这样的(“ ”表示空格,下同):

```
:BLACK 0 FOREGROUND; (黑)
:BLUE 1 FOREGROUND; (蓝)
:GREEN 2 FOREGROUND; (绿)
:LBLUE 3 FOREGROUND; (浅蓝)
:RED 4 FOREGROUND; (红)
:PINK 5 FOREGROUND; (粉)
:YELLOW 6 FOREGROUND; (黄)
:WHITE 7 FOREGROUND; (白)
```

这是 FORTH 中最常用的编程方式,也叫定义。这几行程序,直接由键盘输入即可。当然也可以进入编辑状态,存入文本文件中,然后再编译。两种方式效果都一样,都会使 BLACK、BLUE……WHITE 这几个词进入系统,之后就能根据这几个词改变颜色了。接下来打入

RED

系统会用红色显示,再打入

WHITE

以后显示内容又会变为白色。

这种词可以在程序中被引用。与之类似,还能借助于 BACKGROUND 改变背景颜色。

讲述以上的例子,目的是想使读者通过实例了解 FORTH 语言所具有的扩充能力。对具体操作不理解也不要紧,以后会详细讲到的。这里的定义,又是编译过程,与其他语言相比,具有以下特点:

- \* 每个新功能都要起一个词名,这里是 BLACK、RED、WHITE 等。如何起名,一般没有限制,编程者应该尽量选择含义清楚,便于记忆的名称。

- \* 新词在使用上,格式与原系统一致,一经定义,就可以直接引用这些词,不必理会它们是如何定义的,极为方便。

- \* 新定义的单词,一旦进入系统,就与原有系统浑然一体。系统并不区分哪个词是原有的,哪个词是新定义的。所有单词,在系统中地位都是一样的,实际上,一般的 FORTH 系统,大部分单词都是利用 FORTH 自身的扩充能力而扩充出来的。

- \* 编译过程是一遍扫描,直接在内存中产生目标代码。不象其他语言那样在磁盘中产生中间文件,也不需要表格区,因此非常快速,开销极小。

- \* 产生的目标代码非常简短,经常比汇编还要短,一个系统中可以定义出很多新词,其数量只受内存限制,大多数情况下不必担心系统中存放不下。

- \* 新词一经编译,便进入系统,不必保存源程序。简单的定义可以直接从键盘输入,使系统立即增加新词,随后可以立即使用。



以上介绍了扩充能力,下面再来看看其他特点。

**词典结构。**为了便于使新词放入系统,FORTH 采用了词典结构。系统的主要部分被称为词典,其中是按顺序存放的一个个单词。这些单词相当于其他语言中的命令、语句、函数和子程序等,实际上就是一个个功能模块,系统的功能就是这些词的功能,每个词有一个词名。一般情况下,新词是通过词典中原有词组合而定义出来的,每定义一个新词,就把它放入词典顶端。整个系统就是利用滚雪球的方法建立起来的。反之,又可以用删除顶端一部分单词的方法来缩小词典规模。由于可以方便地扩充和压缩,使得系统非常灵活。可以看出,它是一个开放的系统。词与词之间是由链表连接在一起的,查询、添加、删除等,都要通过这条链,有人把它称为 FORTH 系统的生命线。

词典结构体现了 FORTH 语言的一个原则:简明。它易于理解,进行各种处理也非常方便,易于保证程序的正确性。简明还使得用户分析这个开放系统成为可能,突出的一点,是可以方便地对单词进行反编译。另一方面,它的功能又很强,可以解决各种复杂问题。

**模块化和结构化。**软件工程学倡导模块化和结构化编译方法。这要求所使用的语言具有与之适应的机制,FORTH 语言正是具备了这种机制的语言。词典中一条条单词就是一个个模块,单词构成方法,能够较好地符合模块化所要求的便于分解、隐藏信息和模块独立性。FORTH 又是天然的结构化语言,适宜于自顶向下的设计表达,具体编程时则是自底向上生成各个单词。在程序流程的控制结构方面,FORTH 语言也非常好。它可以方便地实现各种条件判断、循环和分情况结构。如果所需要的结构在一个版本中不存在,还可以由用户自己扩充出来,这点笔者深有体会。值得一提的是,FORTH 中没有 GOTO 这个词,因为不需要。在其他语言,对于 GOTO 使用不当时,会造成非结构化因素,经常是引起程序混乱的祸根,但有时不用 GOTO 却又难以解决问题。FORTH 不存在这样的问题,这与它的高度模块化有关。更值得一提的是,FORTH 中还能方便地容纳与高级成分相一致的结构化汇编,可以省去在普通汇编中必不可少的标号。

**堆栈。**进行模块化编程时有一个非常重要的问题要解决。就是模块之间的参数传递。在其他语言中,一般是利用变量,可能因此带来复杂的问题。很多程序错误,都是因为对变量使用不当引起的。FORTH 没有沿用其他语言中的做法,它使用了堆栈,这是与其他语言的重大区别之一。各种运算、操作,在需要处理和传递数据的地方,一般都是通过堆栈进行的,相应地,采用了一种特殊的运算表示法,称为逆波兰表示法,又叫后缀表示法。堆栈和逆波兰,有其自身的性质,内容很丰富,以后还要重点介绍。

**快速。**随着词典的层层扩充,高层次单词的速度会有所降低。因此词典低层的词应该非常快才行。FORTH 速度非常快,而且还可以将某些对速度有重要影响的词(即所谓“瓶颈”部位的词)用汇编来编写,

以进一步提高速度。

**紧凑。**FORTH 系统占内存很小。在其他语言由于内存紧张难以运行的场合,如各种8位微机、单片机和单板机系统,FORTH 却能舒适地运行。

由于快速和占内存小,在很多场合下,这个语言常常成为取代汇编的首选高级语言。

**集成化环境。**使用 FORTH 时,解释、编译和磁盘存取都在同一个操作界面,使用上统一而方便。编辑器非常小,可以常驻内存,进入和退出编辑都很迅速。加之 FORTH 程序一般比较短,使得编程、调试都很方便,系统兼有解释和编译两种功能,在解释状态下,具有很好的交互功能,能象 BASIC 和 LOGO 那样方便地进行人机对话;另一方面,词典中的词都经过编译,运行效率又非常高。集成化环境,对于其他语言是近年来才发展起来的技术,比较复杂;而对于 FORTH 来说,是早已具备的性能,而且较为简单、自然。

### 发展简史

FORTH 语言是在天文台开发的,最初用于射电天文望远镜的控制,随着美国射电望远镜的出口,流传到了世界各地。由于它的成功,70年代中期,国际天文学会把 FORTH 确定为标准计算机语言。

在最初10年,由于缺乏教科书和它的软件比较昂贵,这个语言传播速度比较缓慢。1978年,一些 FORTH 爱好者成立了一个 FORTH 兴趣小组(FORTH Interest Group 简写 FIG),这是一个非赢利的组织。他们从一个通用模型出发,在当时流行的 8080、8086、6800、6502 和 PDP-11 等9种计算机上实现了 FIG FORTH 版本。由于他们的版本是公开版权的,又易于移植,大大促进了 FORTH 的广泛传播。现在我们国内在 APPLE II 及兼容机上,还流传着这个版本。早几年,宇航出版社的《FORTH 系统》介绍的就是这个版本。PC 机上普遍使用的 PC/FORTH2.0 也是在这个版本基础上发展起来的。

FIG 不断发展,现在已成为遍布世界、会员达十几万的组织。FIG 还很重视语言的标准化工作,先后制定了 FORTH-78、FORTH-79、FORTH-83 等标准。据介绍,现在正在进行新的标准修订工作,不久即将公布。

在国外,FORTH 在天文、军事、航空航天、工业自动化、图形、医疗仪器、工作站等众多领域都有广泛应用。在国内虽不及国外广泛,但也取得了许多可喜的进展。

这个语言引进我国是在1980年,将其汉化,是在中国推广的重要内容,目前已有多种汉化版本。1987年,成立了中国软件行业协会 FORTH 应用分会,简写 CFIG,使 FORTH 语言在我国发展进入了一个新阶段。学习它的人越来越多。1990年3月,机电部仪器仪表司决定将 FORTH 做为仪表行业的标准语言。

由于其自身的特点,加上广大爱好者的努力,可以预计 FORTH 语言还将有一个较大的发展。



# BJS-51单片机实验系统

## 模/数、数/模转换实验(续)

北京工业大学(100022) 张俊谋

### 三、实验操作和编程:

在实验时,应注意以下各项:

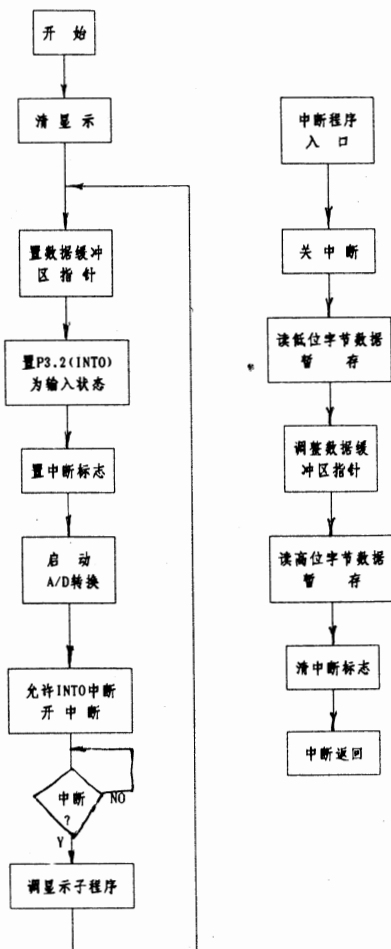
1. 实验前必须根据教师拟定的实验要求,选择相应的插块位置。

例如:在控制转换方式下,模拟信号为4~20mA的电流信号,希望采用中断方式输入数据,测定精度高,拟用外部参考电压。

根据上面的要求,选定插块位置。

- 中断响应方式            W1 置于2-3位置
- 控制转换方式            W2 置于1-2位置
- 直接输出方式            W3 置于2-3位置

框图1:



| 地址   | 机器码    | 行号 | 标号     | 汇编语言             | 注释           |
|------|--------|----|--------|------------------|--------------|
| 8003 |        | 1  |        | ORG 8003H        | 中断0 中断服务程序入口 |
| 8003 | 028280 | 2  |        | LJMP INT01       | 转入中断服务程序     |
|      |        | 3  |        | :                |              |
| 8200 |        | 4  |        | ORG 8200H        | 控制转换, 中断方式采集 |
| 8200 | 9140   | 5  | MAIN1: | ACALL CLR        | 清显示          |
| 8202 | 7820   | 6  | MA12:  | MOV R0, #20H     | 数据缓冲区首地址     |
| 8204 | D2B2   | 7  |        | SETB P3.2        | 设P3.2为输入口    |
| 8206 | D2D5   | 8  |        | SETB PSW.5       | 设置中断标志       |
| 8208 | 907800 | 9  |        | MOV DPTR, #7800H | 启动转换         |
| 820B | F0     | 10 |        | MOVX @DPTR, A    |              |
| 820C | D2A8   | 11 |        | SETB IE.0        | 置外部中断0       |
| 820E | D2AF   | 12 |        | SETB IE.7        | 开中断          |
| 8210 | 20D5FD | 13 | HERE1: | JB PSW.5, HERE1  | 未中断, 等待      |
| 8213 | 7A20   | 14 |        | MOV R2, #20H     |              |
| 8215 | 128400 | 15 | HERE1: | LCALL DISP       | 调显示转换结果子程序   |
| 8218 | 7D01   | 16 |        | MOV R5, #1       |              |
| 821A | 128480 | 17 |        | LCALL DELAY      | 调延时子程序       |
| 821D | DAF6   | 18 |        | DJNZ R2, HERE1   |              |
| 821F | 128202 | 19 |        | LJMP MA12        |              |
|      |        | 20 |        | :                |              |
| 8280 |        | 21 |        | ORG 8280H        | 中断服务程序       |
| 8280 | C2A8   | 22 | INT01: | CLR IE.0         | 禁止外部中断0      |
| 8282 | C2AF   | 23 |        | CLR IE.7         | 关中断          |
| 8284 | 907800 | 24 |        | MOV DPTR, #7800H |              |
| 8287 | E0     | 25 |        | MOVX A, @DPTR    | 读取低位字节       |
| 8288 | F6     | 26 |        | MOV @R0, A       | 暂存数据         |
| 8289 | 08     | 27 |        | INC R0           | 调整缓冲区地址      |
| 828A | 907400 | 28 |        | MOV DPTR, #7400H |              |
| 828D | E0     | 29 |        | MOVX A, @DPTR    | 读取高位字节       |
| 828E | F6     | 30 |        | MOV @R0, A       | 暂存数据         |
| 828F | C2D5   | 31 |        | CLR PSW.5        | 清中断标志        |
| 8291 |        | 32 |        | RETI             | 中断返回         |

- 使用外部参考电压 W4 置于1—2位置
- 4~20mA 电流输入(差分) W5置于1—2位置  
(电流)W6 插上短路块

2. 根据实验要求拟定程序框图和编制软件

(1)根据要求程序如框图1。

(2)根据程序框图,编制的实验程序如表。

3. 本实验中所使用的仪器要求比较高,比如数字电压表宜采用5位半,至少也应保证在四位半的水平。

#### 四、带数据寄存器的8位 D/A 转换总片—DAC0832

1. 芯片特点:

- 分辨率为8位(精确到1/256)

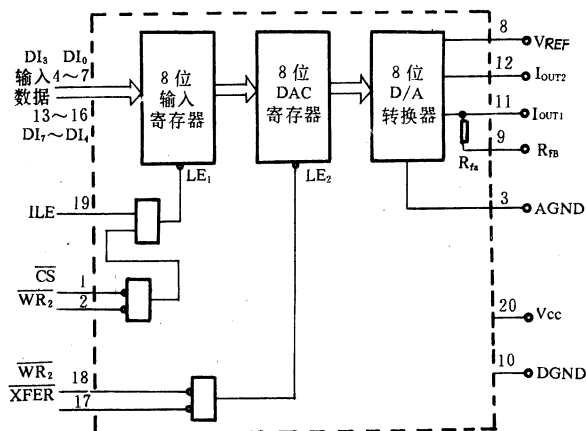


图4

- 电流稳定时间:1 $\mu$ S
- 数字输入可实现双缓冲、单缓冲或者直接输入
  - 只需在满量程下调整其线性度
  - 单一电源供电(+5V~+15V)
- 低功耗:20mW

2. 芯片的结构原理与管脚功能:

DAC0832的原理性结构及外部管脚如图4所示。它由8位输入寄存器、8位 DAC 寄存器、8位 D/A 转换电路及控制电路所组成,为20脚双列直插式封装。

各管脚的功能如下:

- DI0~DI7 8位数据输入端
- ILE 输入寄存器允许信号,高电平有效
- $\overline{CS}$  片选信号,低电平有效
- $\overline{WR1}$  输入寄存器写选通信号,低电平有效
- $\overline{WR2}$  DAC 寄存器写选通信号,低电平有效
- $\overline{XFER}$  传送控制信号,低电平有效。与 $\overline{WR2}$ 一起用来选通 DAC 寄存器
- VREF 基准电压输入端
- Rfb 反馈电阻输入端
- Iout1 电流输出端1,其值随 DAC 寄存器内容线性变化
- Iout2 电流输出端2, Iout1+Iout2=常数
- Vcc 电源电压输入端
- AGND 模拟地
- DGND 数字地

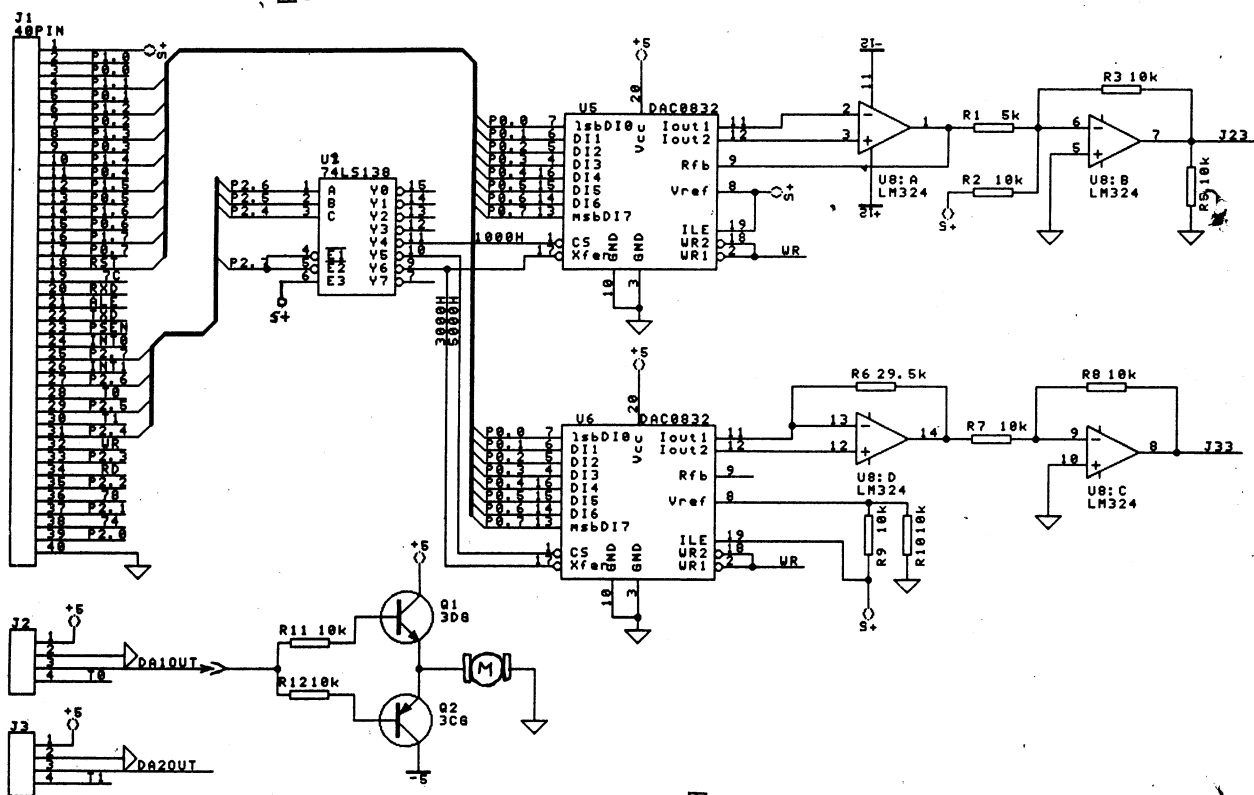


图5

## 五、D/A 转换实验电路的设计

为了满足对 D/A 转换实验的要求,实验电路采用双 D/A 转换器,并且可用于控制微型电动机的启、停和正反转,电原理如图5所示。

该电路包括 D/A 转换器、转换控制电路、输出电路及电机控制电路等几部分。

1. 转换控制电路:这部分由 U2(74LS138)构成,实际上是一个译码电路,由高位地址线 P2.4~P2.7 通过该译码器译出 1000H、3000H、5000H 三个地址供本部分电路使用。1000H 用于控制 U5 的写入寄存器的写入操作;5000H 用于控制 U6 的输入寄存器的写入操作;而 3000H 则用于将两个 DAC0832 输入寄存器中的数据通过 DAC 寄存器送入 D/A 转换电路部分而实现 D/A 的输出。

2. D/A 转换电路:这部分包括两个 DAC0832(U5 和 U6)。从前述控制电路的说明中可以看出,这两个 D/A 转换器采用的是双缓冲的接法,这种接法可以保证两个 D/A 转换器输出同步。

3. 输出电路:这部分包括运算放大器 LM324 和若干电阻。LM324 是一个四运放集成电路。从图5中可以看出 U8:A 和 U8:B 构成 U5(DAC0832)的输出电路。U8:A 是 I/V 转换器,U8:B 是一个偏置电路,使得在插座 J23 上得到  $-5.0V \sim +5.0V$  的双极性输出,用于控制电动机。图中 U8:D 和 U8:C 构成 U6(DAC0832)的输出电路,U8:D 是 I/V 转换器,U8:C 则是一个反相器,使得在插座 J33 上得到  $0 \sim 5V$  的单极性输出。

4. 电机控制电路:这部分包括晶体管  $Q_1$ (3DG)、 $Q_2$ (3CG),电阻  $R_{11}$ 、 $R_{12}$  和电动机 M,由 U8:B 输出的  $-5V \sim +5V$  的电压经晶体管驱动之后,带小电机作正转、反转实验。

### 程序1

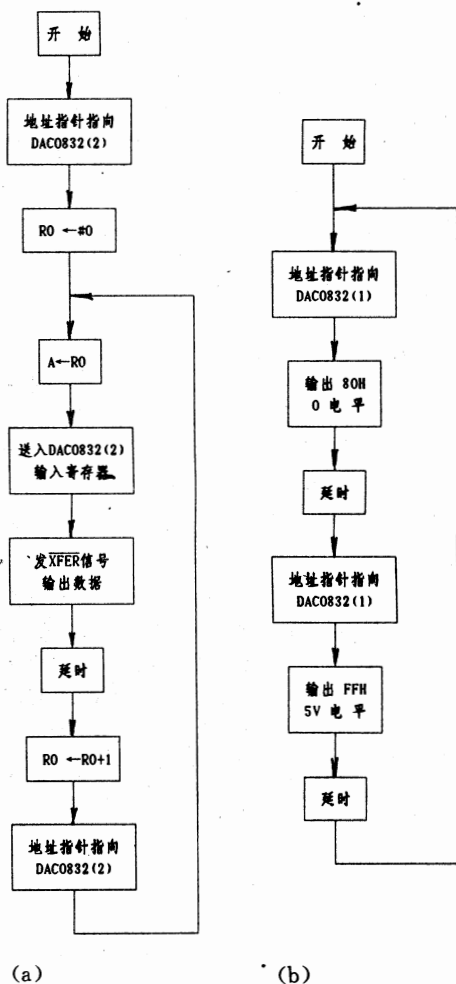


图6 D/A 转换实验程序框图

| 地址   | 机器码    | 行号 | 标号     | 汇编语言             | 注释                |
|------|--------|----|--------|------------------|-------------------|
| 8000 |        | 1  |        | ORG 8000H        |                   |
| 8000 | 4120   | 2  |        | AJMP DASA        |                   |
|      |        | 3  |        | :                |                   |
| 8220 |        | 4  |        | ORG 8220H        |                   |
| 8220 | 805000 | 5  | DASA:  | MOV DPTR, #5000H | 地址指针指向 DAC0832(2) |
| 8223 | 7800   | 6  |        | MOV R0, #00H     | 输出数据从 00H 开始      |
| 8225 | E8     | 7  | WW:    | MOV A, R0        |                   |
| 8226 | F0     | 8  |        | MOVX @DPTR, A    | 送数据到 DAC0832(2)   |
| 8227 | 903000 | 9  |        | MOV DPTR, #3000H | 完成 D/A 转换输出       |
| 822A | F0     | 10 |        | MOVX @DPTR, A    |                   |
| 822B | 128235 | 11 |        | LCALL DELAY      | 延时                |
| 822E | 08     | 12 |        | INC R0           | D/A 转换数据从 00H 开始  |
| 822F | 00     | 13 |        | NOP              | 每次加 1             |
| 8230 | 805000 | 14 |        | MOV DPTR, #5000H |                   |
| 8233 | 4125   | 15 |        | AJMP WW          |                   |
|      |        | 16 |        | :                | 延时子程序             |
| 8235 | 7B02   | 17 | DELAY: | MOV R3, #02H     |                   |
| 8237 | 7CFF   | 18 | T3:    | MOV R4, #0FFH    |                   |
| 8239 | 7DFF   | 19 | T4:    | MOV R5, #0FFH    |                   |
| 823B | DDFE   | 20 | T5:    | DJNZ R5, T5      |                   |
| 823D | DCFA   | 21 |        | DJNZ R4, T4      |                   |
| 823F | DBF8   | 22 |        | DJNZ R3, T3      |                   |
| 8241 | 22     | 23 |        | RET              |                   |



若小电机转动时带动一小叶片,小叶片转动到某一位置时将光电耦合器的光阻断,转速越快,每秒钟阻断光的次数越多,光电耦合器输出的脉冲频率越高,将该脉冲接入8031的T0或T1口,通过编程就可测出电机转速,并可在数码管上显示出来。

#### 六、实验操作和编程:

在上述实验电路的基础上,通过编程将一组数码送入D/A转换器,执行变换指令后可在运放输出端得到相应的一组电压值,如果是按某种规律送入一组数码,则输出电压也将按某种规律变化,如正弦波、三角波、方波等等;如果两路输出均按正弦波变化,且相互间相位差为90°,这样两组电压输入示波器的x,y轴,在示波器上可以观察到圆形或椭圆的轨迹。

程序2

| 地址   | 机器码    | 行号 | 标号       | 汇编语言             | 注释               |
|------|--------|----|----------|------------------|------------------|
| 8000 |        | 1  |          | ORG 8000H        |                  |
| 8000 | 4180   | 2  |          | AJMP DAMOTOR     |                  |
|      |        | 3  |          |                  |                  |
| 8280 |        | 4  |          | ORG 8280H        |                  |
| 8280 | 901000 | 5  | DAMOTOR: | MOV DPTR, #1000H | 地址指针指向DAC0832(1) |
| 8283 | 7800   | 6  |          | MOV A, #80H      | 先送数据80H          |
| 8285 | F0     | 7  |          | MOVX @DPTR, A    | 送数据到DAC0832(2)   |
| 8286 | 903000 | 8  |          | MOV DPTR, #3000H | 完成D/A转换输出0电平     |
| 8288 | F0     | 9  |          | MOVX @DPTR, A    |                  |
| 828A | 518A   | 10 |          | ACALL DELAY1     | 0电平持续时间          |
| 828C | 78FF   | 12 |          | MOV A, #0FFH     | 再送数据0FFH         |
| 828E | 901000 | 11 |          | MOV DPTR, #1000H | 地址指针指向DAC0832(1) |
| 8291 | F0     | 13 |          | MOVX @DPTR, A    | 送数据到DAC0832(1)   |
| 8292 | 903000 | 14 |          | MOV DPTR, #3000H | 完成D/A转换输出电平5V    |
| 8295 | F0     | 15 |          | MOVX @DPTR, A    |                  |
| 8296 | 518A   | 16 |          | ACALL DELAY1     | 5V电平持续时间接触动      |
| 8298 | 4150   | 17 |          | AJMP DAMOTOR     |                  |
|      |        | 18 |          |                  | 延时子程序            |
| 829A | 7B02   | 19 | DELAY1:  | MOV R2, #0FFH    |                  |
| 829C | 7CFF   | 20 | DEL:     | DJNZ R2, DEL     |                  |
| 829E | 22     | 21 |          | RET              |                  |
| 829F | DDFE   | 22 | DELAY2:  | MOVZ R2, #50H    |                  |
| 82A1 | DCFA   | 23 | DEL2:    | DJNZ R2, DEL2    |                  |
| 82A3 | DBF8   | 24 |          | RET              |                  |

如果D/A转换器输出的模拟电压,再送给功率驱动部件执行控制任务,如直流小电机的正、反转及快、慢转等。

如上述可做的实验项目很多,下面仅以输出三角波(锯齿波)和控制直流小电机为例,说明编程以及实验操作方法。

#### 1. 输出三角波实验编程和操作:

根据要求程序框图如图6a所示。

由上述程序框图,编制的三角波实验程序如程序1:

运行上述程序,用示波器在运算放大器LM324(U8,B)的第7脚可以观察到三角波(锯齿波)为了得到清晰的波形,应该适当调节示波器频率开关和同步旋钮。在实验时,改变延时子程序中R<sub>2</sub>预置的数值,可以得到不同斜率的锯齿波。

#### 2. 控制小电机实验编程:

根据控制小电机正、反转和快、慢速的要求,得到程序框图如图6、b所示。

由程序框图编制的实验程序如程序2:

运行上述程序,就可在DAC0832(1)(U5)输出电路的输出端获得一串脉冲,经放大便可驱动小电机移动。

改变程序中语句:828A、828C、8296,可使输出脉冲的直流电平及持续时间变化,从而达到使电机正转、反转,快转和慢转的目的,在不同转动情况下,828A、828C及8296各单元的内容如下表:

| 存储单元 | 正向快转 | 反向快转 | 正向慢转 | 反向慢转 |
|------|------|------|------|------|
| 828A | 519A | 519A | 519F | 519F |
| 828C | 74FF | 7400 | 74FF | 7400 |
| 8296 | 519A | 519A | 519F | 519F |

#### 结束语:

到本期为止,有关BJS-51单片机实验系统系列讲座已经结束。我们将在“电子与电脑”杂志的帮助下,将讲座的内容汇集成册出版。研制这样一个有关单片机的教学实验系统是我们的初次尝试,不足和错误一定不少,敬请广大读者赐教。有意见和愿购买该教学实验系统的同志请与北京市单片机应用技术协会技术服务部盛志纯同志联系。地址:北京西城三里河西口18号楼,邮编100045。



# 中华学习机巧测电容

四川重庆市20信箱62分箱 邓本富

我们知道,在 CEC—I 中华学习机上有一个九针游戏杆插座。这个插座可提供4个模拟量和3个开关量的输入,用 BASIC 语言中的 PDL(n)函数可以读取 n 号(n=0~3)摇杆的编码值(0~255)。实际上,该编码值就是学习机内部定时器组件的延时循环次数。机内监控程序的循环时间为11个时钟周期,共计10.8 $\mu$ S。如果我们在模拟量输入端接入被测电阻,用一个简单的 BASIC 程序调用监控程序,测出延时循环时间 t(t 等于10.8 $\mu$ S 乘以延时循环次数),通过 RC 电路充电时间关系式  $V_c = V_{cc}(1 - e^{-t/RC})$ ,即可算出被测电阻的阻值。关于这一点,清华大学出版社出版的《苹果机和中华学习机应用实验与实用制作》一书中已有详述,本文不再赘述。

在《实用制作》一文中,对测量电容须增加接口电路,由于接口电路要用 STB 选通信号控制工作,而 STB 信号输出只有 APPLE—I 机上的16脚游戏 I/O 接口才有,在 CEC—I 机上无此选通信号输出,所以无法实现。而笔者通过分析认为,在 CEC—I 上,不用增加接口电路,完全可以测量电容。下面谈谈测量方法。

电容的测量原理和测量电阻类似,仍然利用 RC 电路充电的时间关系。CEC—I 中用 NE558(U19)作为定时器组件,其内部由4个独立的555时基电路所组成,图1示出其四分之一,相当于一路模拟量输入接口(对应 PDL0)。虚线为测量电容时所加。电阻  $R^*$  和电容  $C(C = C_1 + C^*)$  构成时基电路 RC 充电回路。当触发输入端(3脚)接到一个负向脉冲时,输出端(1脚)输出为高电平,负脉冲消失后,电源电压  $V_{cc}$  通过  $R^*$  对 C 充电,2脚电压逐渐上升,当2脚电压超过  $2/3V_{cc}$  时,时基电路迅速翻转,1脚输出变为低电平。只要测出1脚高电平的持续时间,通过算式  $t \approx 1.1RC$  即可得到 C 的容量值, $C^*$  的具体计算公式为:

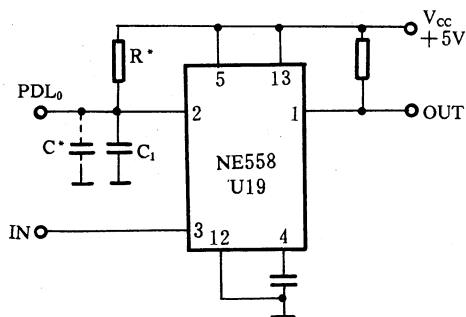


图1

$$C^* = 10.8M / 1.1R^* - 0.022(\mu F)$$

注:式中  $R^*$  的单位为欧姆

测量电容的 BASIC 程序清单如下:

```
10 DIM A(20)
20 K=0
30 FOR I=1 TO 20:A(I)=PDL(0);K=K+A(I);NEXT I
40 M=K/20;IF M>=255 THEN PRINT"CONDENSER TOO LARGE":PRINT"CHANGE THE RESISTANCE";END
50 C=10.8*M/1.1*R-0.022
60 PRINT"C=";C;"mF"
70 FOR I=1 TO 3000:NEXT I;GOTO 20
```

测量时,连接方法如图2。把  $R^*$  接入+5V 和 PDL(0)端, $C^*$  接入 GND 和 PDL(0)端,运行 BASIC 程序即可。若换接不同的  $R^*$ ,则  $C^*$  的可测范围亦随之变化。 $R^*$  越大,可测  $C^*$  的最大值越小,但分辨率提高。

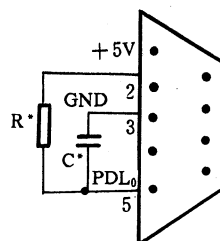


图2

下面是  $R^*$  与  $C^*$  的大致关系

| $R^*$ | $C^*_{min}$    | $C^*_{max}$ | 分辨率           |
|-------|----------------|-------------|---------------|
| 10k   | 0.0006 $\mu$ F | 0.2 $\mu$ F | 0.001 $\mu$ F |
| 1k    | 0.03 $\mu$ F   | 2 $\mu$ F   | 0.01 $\mu$ F  |

这种电容测量方法,经笔者在中华学习机上实践,效果良好,有兴趣的读者不妨一试。

# 家庭如何配置中华学习机

上海交通大学闵行二部(200240) 聂铁轮

1. 首先应当清楚,购买学习机用来完成何种任务。根据不同的要求,购买适当的学习机及其软件。

CEC-M型是一种初级的中华学习机,不能显示彩色图案,不能外接驱动器,其内固化BASIC解释程序,局限于对儿童进行普及型教育,其工作范围有限。

CEC-I型,这种计算机对于家庭比较适用,它不仅与Apple I型计算机完全兼容,而且具有汉字输入显示功能,可外接软盘驱动器,CEC-I型的汉字功能已经固化在计算机的主机板上,可随时输出汉字,显示器的接口可接PAL制式彩色或黑白电视机,另外还设有游戏控制杆接口。

2. 对CEC-I型功能的扩充,及外围设备的配置:

在CEC-I型机主机板上有一个与Apple I兼容的输入、输出插座,用来扩充CEC-I型机的功能。

1. Z-80卡:由于CEC-I型中华学习机使用的操作系统是DOS系统,所以在CP/M操作系统下,数以千计的软件无法使用,我们可以在扩充槽口上插入Z

-80卡,在Z-80卡中有固化的CP/M操作系统,就可以使用CP/M操作系统下的大量软件,Z-80卡售价160元左右。另外还需一台软盘驱动器。

2. 使用dBASE-II系统,如果我们想在CEC-I上建立dBASE-I管理系统,至少需要56K的内存,CEC-I型机内存为69KRAM,可以满足需要。由于dBASE-I管理系统是用8080汇编语言编写的。需在扩充槽口插入Z-80卡,再购买一台软盘驱动器,将含有dBASE-I的软盘调入内存,就可使用了。有时我们需要把汉字库调入内存,使用汉字,如果内存中存在汉字库和dBASE-I管理程序,可能有些内存的字节溢出,发生这种情况,就要在扩充槽上插入扩充卡,如16K的RAM等。

3. 如果你要在屏幕上进行汉字编辑并打印编辑内容,可以购买CPC卡,此卡有汉字编辑和打印机驱动功能,使用很方便。如果仅仅局限于打印,可购买打印卡,接在扩充槽上。

## 92'全国软件出版工作研讨会 在湖北宜昌召开

1991年10月15日至20日,由中国软件行业协会软件出版分会与湖北省新闻出版局联合举办的“92'全国软件出版工作研讨会”在宜昌市召开。国家新闻出版署与机械电子工业部对会议的召开给予了大力支持。国家新闻出版署刘臬副署长亲笔至函祝贺,并对软件出版工作表示极大的关注;技术发展司、图书司、版权司的有关领导出席了会议,宜昌市的党政领导到会祝贺。《计算机世界》报打来电报预祝会议圆满成功。来自北京、上海、天津、华东、东北、西北、西南、华中等地区的出版社、出版管理部门、软件开发单位以及有关报刊等36个单位的48位代表参加了会议。湖北省新闻出版局副局长田胜立同志与软件出版分会秘书长宋玉升同志主持了会议。

新闻出版署技术发展司司长高永清同志在大会上对于近年来电子出版物的发展作了重要讲话。

这次研讨会共收到论文21篇,大会报告16篇,对于国内外软件出版概况,软件出版工作流程及文档管理,软件的选题与组稿,软件编辑岗位责任制问题,软件出

版物的质量与商品化,软件的版权保护问题以及软件的发行与软件市场等进行了深入的研究与讨论。代表们一致认为,软件出版是我国软件产业中不可忽视的一支力量,工作取得了长足的进步,但是还存在许多问题和困难,例如,软件出版的法律效力不明确,发行渠道不流畅,软件编辑人员的工作还需要有关领导的确认和大力支持。

软件出版在我国已初具规模,开拓了新的出版业务,国家新闻出版署极为关心软件出版事业的发展。在当前国际上电子出版物发展迅速并处于激烈的竞争的情况下,应进一步加强对包括计算机软件、数据库、电子书刊在内的电子出版物的规划、领导与管理,这对于我国电子出版事业的发展,具有重要的意义,特别是要研究解决软件以国际标准书号出版的法律效力及今后电子出版物的版权保护问题,制定一系列软件出版工作规范与管理条例,加强编辑队伍建设,协调软件发行工作,把我国软件出版事业推进到一个新阶段。



# F BASIC 语言的游戏程序编程技巧

## 第四讲 游戏程序的设计过程(中)

山东苍山机械电子化学工业局(277700) 于春

(6)模块 H:

成龙若按 PS\$ 中记录的顺序打开箱子则发声,打印积分,并令箱子消失。每打开一个箱子加10分。若箱子全部打开则转救人成功程序。流程图见图5:

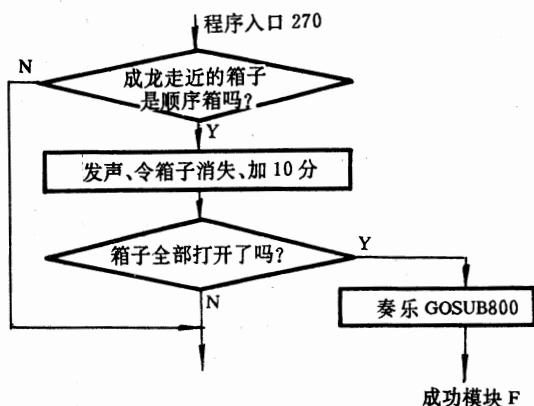


图5 模块 H 的流程图

```
270 REM "OPEN BOX" (打开箱子)
275 IF S1$ <> ML (PS$, PS, 1) T. 295
280 PL. "O4C0#CD#DEF#FG#G"
285 LOC. PX+SX, PY+SY; P. "┌";
    T=T+10; LOC. 8, 22; P. T
290 PS=PS+1; IF PS>PL T. PL. "O4C3D
    EFG; O4E3 FGAB"; G. 110 (成功处理)
295 RE.
```

注:┌表示空格,下同

(7)模块 I:

妖怪自动运动。轮流测试妖怪在 BG 面的座标。判断妖怪周围是否为楼板或楼梯,使妖怪在楼板上左、右移动,在楼梯上上、下移动。使用变量:

KD:妖怪的运动方向。

F:妖怪动作的编号。KN=2时, F=0,

1KX, KY:妖怪在 BG 面的座标。

JX, JY:成龙在妖怪上下左右的判别。

SK\$, SR\$, SL\$, SD\$, SU\$:妖怪

前后左右的背景数值。流程图见图6。

对应程序如下:(程序入口300)

```
300 REM "BOGY MOVE"(妖怪运动)
```

```
305 F=F+1; IF F>KN-1 T.F=0
310 KD = VCT(F); IF KD>0 T. 320
315 KD=E(F)
320 KX=(XPOS(F)-12)/8; JX=SGN(PX-KX)
325 KY=(YPOS(F)-16)/8; JY=SGN(PY-KY+1)
330 SK$=SCR$(KX, KY); SR$=SCR$(KX+1, KY);
    SL$=SCR$(KX-1, KY); SD$=SCR$(KX, KY+1);
    SU$=SCR$(KX, KY-1)
335 IF (SR$+SL$ <> "┌") AND (SD$+SU$ <> "┌")
    T. 345
340 IF (KD=3 AND JX=-1) OR (KD=7 AND JX=1) OR
    (KD=5 AND JY=1) OR (KD=1 AND JY=-1) T. 390
342 KD=(KD+4)+8*(KD>3); G. 390
345 IF JX * JY <> 0 AND RND(2)=1 T. 355
350 IF JX=1 AND SR$ <> "┌" T. KD=3; G. 390
352 IF JX=-1 AND SL$ <> "┌" T. KD=7; G. 390
```

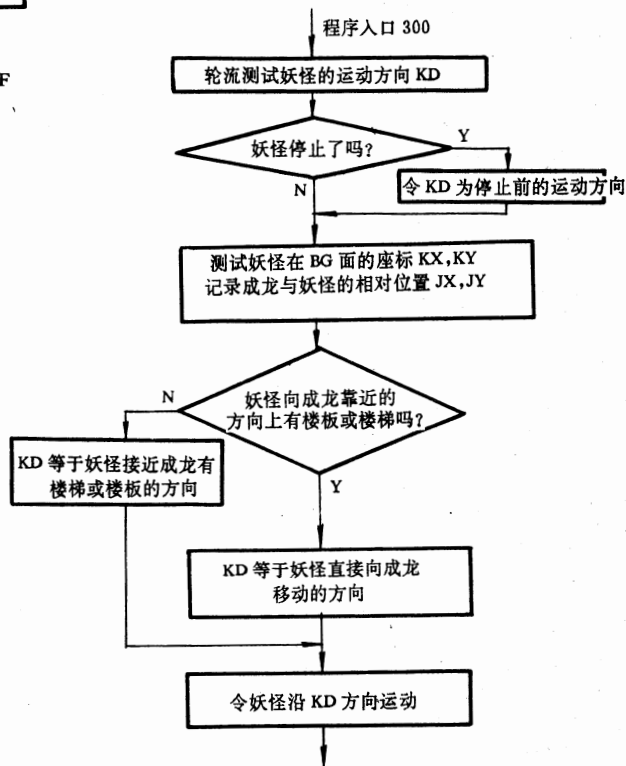


图6 模块 I 的流程图

```

355 IF JY=1 AND SD$ <> "┐" T. KD=5; G. 390
360 IF JY=-1 AND SU$ <> "┐" T. KD=1; G. 390
365 S=RND(4)+1; ON S GOTO 370,375,380,385
370 IF SR$ <> "┐" T. KD=3; G. 390
375 IF SL$ <> "┐" T. KD=7; G. 390
380 IF SD$ <> "┐" T. KD=5; G. 390
385 IF SU$ <> "┐" T. KD=10
390 DE. M. (F)=SP. (11, KD, 1, 4, 0, RND(4)); POS. F, KX
    * 8+12, KY * 8+16; M. F
395 E(F)=KD; RE.

```

#### (8) 模块 F

救人成功处理。令所有定义的卡通消失。画室内地板。定义成龙、金凤相向奔跑、拥抱。成龙起始座标(0, 104), 金凤起始座标(240, 104)。音乐停止后, 清屏、画路。成龙、金凤手拉手向屏幕左方运动(回家)。成龙起点座标(220, 104), 金凤起点座标(230, 104)。流程图见图7, 对应程序清单如下:

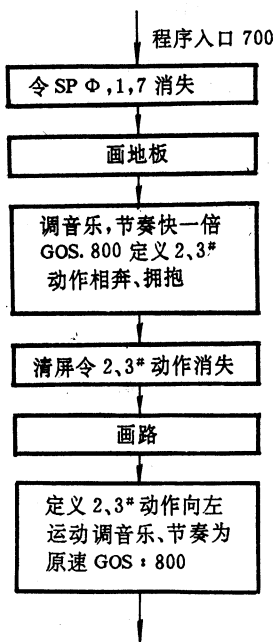


图7 模块 F 的流程图

```

700 REM "VICTORY1" (救人成功)
705 ERA 0, 1, 7; CLS; SP. 3
710 F. I=0 TO 27; LOC. I, 12; P. CH. (208); N.
715 PAL. B 0, 10, 10, 21, 49
720 F. I=2 TO 3
725 DE. M. (I)=SP. (I-2, 4 * I-5, 2, 58, 0); POS. I, (I-2)
    * 240, 104; M. I; N.
730 PL. "M0V12M1V9Y2T1; M0V12T1; M1V0T1"; FF=0;
    GOS. 800
750 REM "VICTORY2"
755 ERA 2, 3; CLS; PAL. B 0, 5, 22, 39, 56
760 F. I=0 TO 27; LOC. I, 12; P. CH. (197); N.
765 F. I=2 TO 3

```

```

770 DE. M. (I)=SP. (I-2, 7, 4, 230, 0)
775 POS. I, (I-3) * 8+230, 104; M. I; N.
780 PL. "M0V12M1V12Y1T4; M0V9T4; M1V2T4"; FF=1;
    GOS. 800
785 RE.

```

#### (9) 模块 K

结束处理, 若是成龙被妖怪抓住或者时间到则打印游戏结束, 询问是否继续游戏; 若救人成功, 在显示成功画面后, 直接询问是否继续。两种情况用一个选项程序, 即读1#操纵器的 START 钮和 SELECT 钮。若继续游戏按 START 钮; 否则按 SELECT 钮。模块流程图和程序清单见管理程序。

#### (10) 模块 J:

音乐模块。为简化程序, 成龙与金凤散步音乐、每

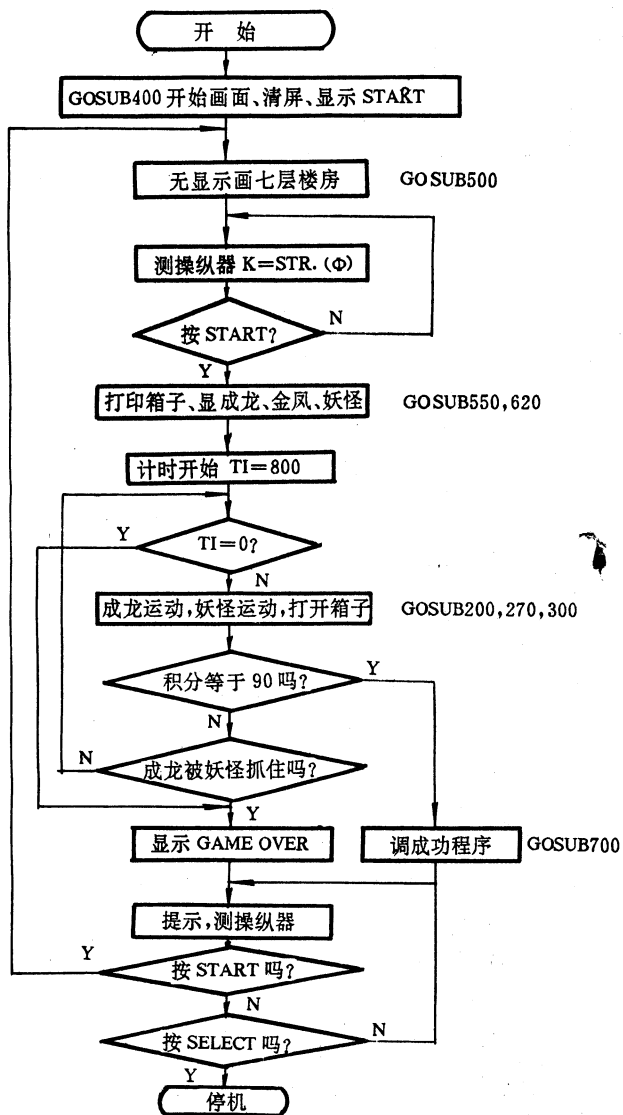


图8 管理程序流程图



```

10 REM"MAIN PROGRAM" (主程序)
15 GOS. 400 (开始画面)
20 CLS:CLS:CG. RND(2),RND(3);CGEN2
25 LOC. 8,8;P. "START!!!"
30 SC. 0,1;PL. "M0V12M1V12Y1T3;M0V9T3;M1V2T3";
   FF=1;GOS. 800
35 SP. O.:PL. "V15T1;V15T1"
40 IF SCR$(1,2)<>" " T. 50
45 GOS. 500 (画七层楼)
50 SC. 0,0;V.:P. " " TIME = " " " " " " " " PRESS "
   START"
55 K=STR.(0);IF K=0 T. 55
60 PL. "O3C1EE;O3E1FG"
65 GOS. 550 (显示九个箱子)
70 GOS. 620 (显示成龙、金凤、妖怪)
75 TI=TI-1;LOC. 7,0;P. TI;" " " " " " (计时开始)
80 IF TI=0 T. 115
85 GOS. 200 (用操纵器控制成龙)
90 GOS. 270 (打开箱子的处理)
95 GOS. 300 (妖怪自动运动)
100 IF CR.(7)>=0 T. PL. "O1B2AGFEDC";G. 115 (救
   人失败)
105 G. 75
110 GOS. 700;G. 130 (救人成功)
115 CLS;LOC. 9,6;P. "GAME"OVER!"
120 SP. OF.
130 LOC. 2,20;P. "CONTINUE/start"END/select"
135 S=STR.(0);IF S=0 T. 135
140 IF S=1 T. RUN 20
145 IF S=2 T. CLS;LOC. 10,4;P. "GOOD-BY!";LOC. 0,
   21;E.
150 G. 135

```

现在“成龙救金凤”游戏程序全部编写完毕。把各模块程序合成即得模块结构程序清单。读者可自己合成。

```

860 REM"MUSIC1"
805 PL. "O4G5E3GA4G1;O4C5O3GA;O1C3CC5C3C"
810 PL. "E3DCDE4D1;GDG;C5C3CC5"
815 PL. "E3 G7G3E4G1;AGGO4C;C3CC5C3CC5"
820 PL. "A3O5C6O4G3A;DCO3G;C3CC5C3CC5"
825 PL. "DE1DC7;CEC;C5C3CC5"
830 PL. "G5E3GA4G1;O4CO3GA;G3GG5G3G"
835 PL. "E3DCDE4D1;GDG;G5G3GG5"
840 PL. "E3G7G3E4G1;AGGO4C;G3GG5C3CC5"
845 PL. "A3O5C6D3G;ECD;C3CC5C3C"
850 PL. "DE1DC7;GDC;C5C3CC5"
855 IF FF=0 THEN FF=1;G. 805
860 RE.

```

900 REM "MUSIC2"  
905 PL. "M0V10M1V8Y3T2;M0V3T2;T2"  
910 PL. "O4C3CO3GA;O3C3CO2GA;O2C5G"  
915 PL. "O4CCO3GADG6;O3CCO2GADG6;CGDG"  
920 PL. "D3G6A3O4E6;D3G6A3O3E6;DGAO3E"  
925 PL. "O3A3O4E6;O2A3O3E6;O2AO3E"  
930 PL. "C3CO3GA;C3CO2GA;O2CG"  
935 PL. "O4CCO3GA;O3CCO2GA;CG"  
940 RE.

管理程序完成整个游戏过程的控制。其流程图见

管理程序如下：

应广大读者要求,我杂志明年新增“IC 与应用”、“电脑与通信”两栏目。“读者联谊”栏目将调整为几个小栏目,有‘读者信箱’、‘购机指南’、‘求答’、‘转让’等,以便更好地为读者服务。欢迎大家踊跃投稿。内容范围大致为:有关国内外 IC 的发展及最新 IC 的应用资料,如开关电源(图像处理、模糊逻辑、通讯)IC 等。有关点对点及网络通信,办公室自动化方面。字数一般在5000字以下。稿件一经利用,即付稿酬。

来稿请寄：北京173信箱杂志编辑部收

邮编:100036 (请在信封上注明“IC 与应用”)

## 敬告读者

本刊92年第3期登出的 PAL 卡,因卡上的调试点需要用户自行调整后才能达到预期效果。

为此购买者请与编辑部直接咨询。本刊  
93年撤消七号信柜邮购点。

《电子与电脑》编辑部



# LQ-1600K 打印机接口专用集成电路 M54610P 的简易修复方法

陕西省计量测试研究所(710048) 赵继文

## 1. 专用集成电路 M54610P 简介

LQ-1600K 打印机并行接口电路由一片专用集成电路 M54610P 构成。其内部逻辑电路如图1所示,各引脚功能列于下表中。

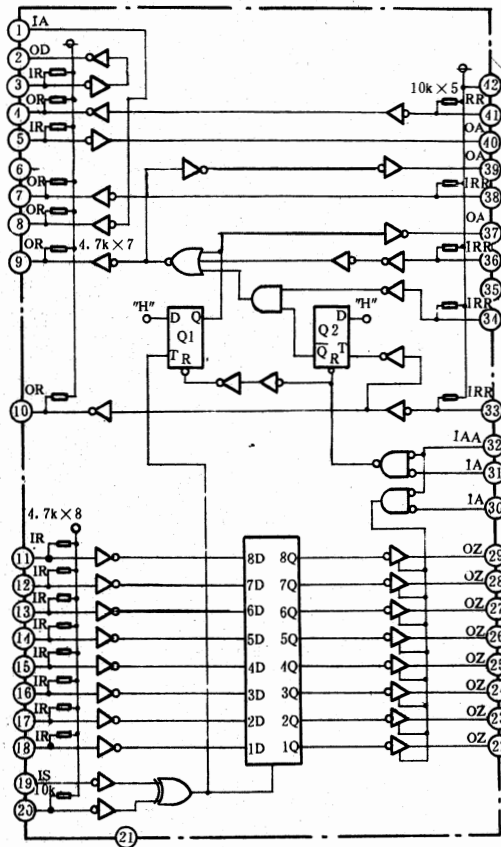


图1

## 2. M54610P 的一种简易修复方法

针式打印机自检正常,而联机不打印或打印错码,都是由于接口电路损坏或接口电缆连接器产生故障而造成。本文介绍 LQ-1600K 打印机并行接口电路损坏的一种简易修复方法。其故障现象是:开机自检正常,但不能联机打印,缺纸灯常亮,实际上并不缺纸。经用万用表测试 CN1(与计算机相连接的接插件的第12针,它的输出表示 PE(缺纸)信号,测试结果为常高电

M54610P 引脚功能表

| 引脚号 | 名称                                                           | 信号方向     | 功 能                                               |
|-----|--------------------------------------------------------------|----------|---------------------------------------------------|
| 1   | BI1                                                          | 入        | PE(缺纸信号)                                          |
| 2   | BO1                                                          | 出        | 送往复位电路的 $\overline{\text{INIT}}$ 信号               |
| 3   | $\overline{\text{BI2}}$                                      | 入        | 来自主计算机的 $\overline{\text{INIT}}$ 信号               |
| 4   | $\overline{\text{BO5}}$                                      | 出        | 来自主计算机的 $\overline{\text{ERROR}}$ 信号              |
| 5   | BI3                                                          | 入        | READY(准备就绪)信号                                     |
| 6   | GND                                                          | —        | 地                                                 |
| 7   | BO4                                                          | 出        | 送往主计算机的 PE (缺纸)信号                                 |
| 8   | $\overline{\text{BO1}}$                                      | 出        | 送往操作面板的 $\overline{\text{PELED}}$ (缺纸灯)信号         |
| 9   | BUSY                                                         | 送往主计算机   | BUSY(忙)信号                                         |
| 10  | $\overline{\text{ACK}}$                                      | 出        | $\overline{\text{ACKNLG}}$ (应答)信号                 |
| 11  | DIN8<br>12<br>13<br>14<br>15<br>16<br>17<br>18<br>DIN1       | 由主计算机送入  | 8位并行数据                                            |
| 12  |                                                              |          |                                                   |
| 13  |                                                              |          |                                                   |
| 14  |                                                              |          |                                                   |
| 15  |                                                              |          |                                                   |
| 16  |                                                              |          |                                                   |
| 17  |                                                              |          |                                                   |
| 18  |                                                              |          |                                                   |
| 19  | STB                                                          | 由主计算机送入  | STROBE(选通)脉冲                                      |
| 20  | PSW                                                          | 由 CPU 送入 | BUSY 信号输出时间<br>低电平;STROBE 的上升沿<br>高电平;STROBE 的下降沿 |
| 21  | GND                                                          | —        | 地                                                 |
| 22  | DOU1<br>DOU2<br>DOU3<br>DOU4<br>DOU5<br>DOU6<br>DOU7<br>DOU8 | 由 CPU 送入 | 8位并行数据                                            |
| 23  |                                                              |          |                                                   |
| 24  |                                                              |          |                                                   |
| 25  |                                                              |          |                                                   |
| 26  |                                                              |          |                                                   |
| 27  |                                                              |          |                                                   |
| 28  |                                                              |          |                                                   |
| 29  |                                                              |          |                                                   |
| 30  | RD                                                           | 由 CPU 送入 | $\overline{\text{READ}}$ (读)脉冲                    |
| 31  | $\overline{\text{WR}}$                                       | 由 CPU 送入 | $\overline{\text{WRITE}}$ (写)脉冲                   |

|    |                   |          |                                                                                    |
|----|-------------------|----------|------------------------------------------------------------------------------------|
| 32 | $\overline{CS}$   | 由 G/A 送入 | Chip select(片选)信号                                                                  |
| 33 | $\overline{ACKI}$ | 由 CPU 送入 | Acknowledge(应答)信号                                                                  |
| 34 | BSSL              | 由 CPU 送入 | BUSY 信号的选择<br>低电平:当 ACKI 变高时,<br>BUSY 信号变为低;<br>高电平:当 CS 和 WR 都变低<br>时,BUSY 信号变为高。 |
| 35 | GND               | —        | 地                                                                                  |
| 36 | EBSY              | 由 CPU 送入 | BUSY(忙)信号                                                                          |
| 37 | BSYF              | 送往 CPU   | BUSY(忙)脉冲                                                                          |
| 38 | BI4               | 由 CPU 送入 | $\overline{PE}$ (缺纸)信号                                                             |
| 39 | RDY               | —        | 无用                                                                                 |
| 40 | BO3               | 出        | 送往操作面板 RYLED<br>(准备就绪指示灯)信号                                                        |
| 41 | $\overline{BI5}$  | 入        | 来自 CPU 的 ERROR<br>(出错)信号                                                           |
| 42 | Vcc               | 入        | +5V 电源                                                                             |

平。该信号来自 M54610P(代号5A)的引脚7,从图1可看出引脚7的内部是一只独立的反相门电路,7脚是反相门电路的输出端,38脚是输入端,它的输入信号来自用作 CPU 的  $\mu$ PD7810HG(代号7B)。再测38脚,结果也为高电平,说明 CPU 送来的信号是正常的,确认反相门电路损坏。与 PE(缺纸)信号及  $\overline{PELED}$ (缺纸指示灯)信号有关的电路用图2表示。由于输入、输出端分别为38和7引脚的反相门 Q16损坏,虽然 CPU 送出的信号为正常的高电平,但经 CN1的第12针反映到主计算机上的 PE 信号为有效的高电平,所以不能联机打印;

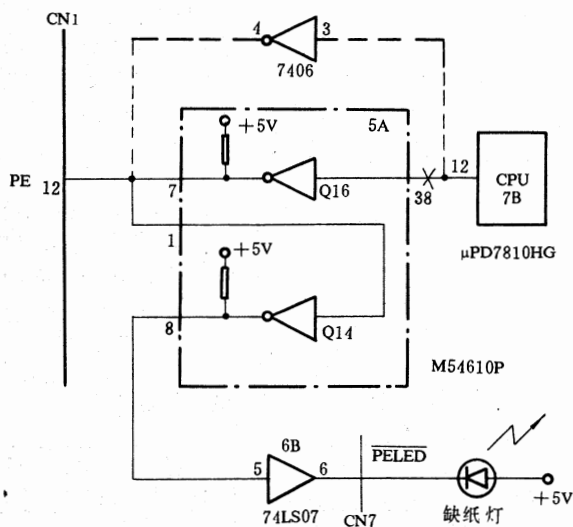


图 2

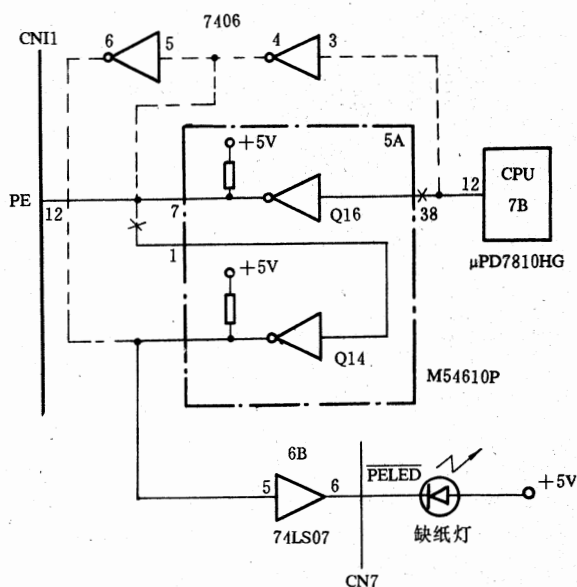


图 3

从图2可以看到5A的引脚1和引脚7相连接,由于引脚7是高电平经反相门 Q14后,从5A引脚8上输出低电平,经74LS07(代号6B)后仍输出有效的低电平缺纸指示灯信号  $\overline{PELED}$ ,所以缺纸灯常亮。按理应用 M54610P 更换5A,但考虑到 M54610P 专用集成电路价格昂贵,且市场上也不易购到,故采用外加一片7406来替换5A中所损坏的反相门电路 Q16。具体方法是:在主板电路板上将5A的引脚38与7B的引脚12的印制连线割开,将7406的引脚3接到7B的引脚12上,7406的引脚4接到5A的引脚7上,用原 Q16输出端的上拉电阻作为7406的上拉电阻。在图2中用“×”号表示要割开的点,用虚线表示要连接的线。并将7406的引脚14和7分别接到+5V和地线上,其余引脚向上翻起,以避免造成短路。用这种简易可行的方法修复 M54610P 后,打印机故障立即排除。

其所以会损坏接口集成电路,绝大多数是由于用户带电插拔接口连接器,或带电拨动机械式多路转换器造成。因为这种带电操作在连接器与接口集成电路相连接的各个端口都会产生突发性冲击电流,若冲击电流过大,集成电路质量又欠佳时,就会损坏电路。本文前述的现象是反相门 Q16损坏。实际工作中也可能反相门 Q16和 Q14都同时损坏,损坏后在5A的引脚8上也始终为高电平,此时经同相门6B后仍为高电平。这时缺纸灯虽不亮,但由于5A的引脚7是常高电平,故打印机仍将产生不联机打印的故障。若遇到这种情况时,可用7406芯片中另外一只反相门来替换 Q14,具体方法见图3,除采取前述分割、连接步骤外,还将7406的引脚4和5串联起来,引脚6连接到5A的引脚8上,再将5A的引脚1和8之间的印制线割开即可。

# GW286计算机硬盘控制卡

## 故障处理一例

湖南黔阳县委办公室(418100) 杨远成

故障现象:开机后,机器不能由硬盘启动,用DOS软盘可启动系统,但不能进入硬盘,并显示:“无效的驱动器标识符”。

故障分析:首先,按Ctrl+Alt+Esc键,检查CMOS RAM系统配置参数均为正常。然后在A驱动器运行GW286诊断程序,对硬盘进行诊断时屏幕显示“1714 DRIVE NOT READY FIXED DISK DRIVE C”。即硬盘驱动器未准备好,无“准备好”信号。通过对以上故障现象分析,可能有以下两种原因,一是硬盘本身的问题,主要是硬盘0磁道信息丢失或硬盘0磁道损坏;二是硬盘控制卡或驱动器等方面的问题。

处理方法:首先检查第一种情况,使用FDISK进行硬盘DOS分区,运行失败,并显示:“Error reading fixed disk”。再进行低层格式化也失败。由此可排除产生第一种情况的可能性。接着对第二种情况进行检查,关机后打开主机机壳,取出硬盘控制卡,更换该控制卡,开机,硬盘能够正常启动进入系统。又经检查,原硬盘控制卡也没发现什么问题。再次关机,重新装回原硬盘控制卡,并将各连接部件安装牢固,使其接触可靠,再重新开机,机器由硬盘引导启动成功,且原硬盘上全部信息完好无损,故障得以排除。由此分析可见,该故障为硬盘控制卡及其连接电缆接口部位松动或接触不良所致。

## TP801单板机的常见故障

江苏邳县港务局第一港埠公司中控室  
(221300) 花成

TP801单板机是目前国内应用比较普遍的一种以Z-80为CPU的微机,由于其性能好,指令丰富、功能强,可扩展性好,集成芯片较通用等特点,而被广泛应用于各工科院校计算机原理实验机,工业控制采样机,

成为实现智能化的得力助手。但由于单板机在使用时比较频繁,因此总会出现一些常见故障,使用者应能分析排除这些常见故障,使之能正常运行工作,下面对TP801单板机中一些常见故障作简要分析:

### 一、电源故障

TP801单板机工作在+5V电压下,假如把电源接在+15V或把电源极性接反,则在工作时,数码显示管出现忽有忽无,时强时弱的不稳定信号,更严重的是,这种人为故障特别容易损坏CPU芯片,造成整机无法工作。

### 二、2716EPROM 损坏

2716EPROM芯片内固化了许多监控程序,当出现故障时,会出现监控失调,比如:正在输入程序时,会突然莫名其妙地跳到其他单元或出现其他错误信息,排除该故障的方法之一便是更换一块新的2716EPROM芯片,这是最保险的一种办法。

### 三、2114RAM 损坏

由于TP801单板机主要是对内存操作,因此,2114RAM芯片损坏是一种最常见的故障,判断该种芯片是否损坏的方法有:

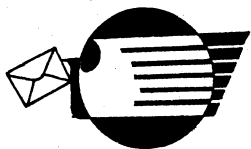
1. 在用户读写存储区内用NEXT键或LAST键进行存写或读取,当发现某一单元无法读写或出现混乱时,则一般情况下,是RAM芯片损坏。例如低四位或高四位数据始终不能改变,而其它RAM区却正常读写,单板机在未损坏的区域内仍能执行各种功能。这种故障排除很简单,只要根据译码电路和出错单元号更换一块新的RAM芯片即可。

2. 该种情况下单板机根本无法进行,数码显示管内容出错或根本没有显示。碰到这种情况,人们往往误认为是CPU或显示电路部分出了故障,但这时最容易出故障的却是地址2C00—2FFFH的RAM区,因为这一区域(1K)中有监控程序使用的暂存区,用户堆栈工作区,用户程序寄存器存放区等,如果这1KRAM出了故障,势必使整个单板机无法运行,所以当出现此种故障时,应该仔细考虑这个问题。

### 四、MON键或STEP键失效

MONITOR监控键及SINGLE STEP单步执行键是TP801单板机上最重要的两个功能键。如果这两个键都失效的话,则很有可能是Z80-CTC出了故障。因为MON键和STEP键要实现其正常功能,必须通过Z80-CTC向CPU请求屏蔽中断不可。如果STEP键正常,而MON键失效,则损坏的可能性最大的是MC14538单稳态触发器,因为MON键是通过此单稳态触发器向Z80-CTC发送计数脉冲的。

上面所述的只是TP801单板机的几种常见故障,单板机一般用于现场控制,容易受干扰,只有正确合理地维护好单板机,才能减少单板机的故障。



## 普及型 PC 个人用户软件交流联谊活动 问题解解答(十二)

### 读者联谊

北京中国农科院计算中心(100083) 王路敬

#### 49. 硬盘的主引导扇区数据破坏后不对硬盘进行低级格式化能否恢复?

不管 IBM-PC 还是兼容机一些意外事故造成硬盘数据丢失,自举失败,从软盘启动也无法进入硬盘,如果立即对硬盘进行格式化处理,会造成大量数据损失,在上述故障中其中很重要的原因之一有可能硬盘的主引导扇区数据被破坏。这时不要动不动就对硬盘进行低级格式化处理,可以将一个相同微机系统中硬盘上的主引导扇区读出来写在软盘某一扇区,然后再将写入软盘这一扇区的主引导扇区的信息写回被破坏的硬盘主引导扇区。可用下面汇编小程序实现之。

程序一 正常硬盘引导扇区的内容写入软盘

```
A)DEBUG
-A 100
3356:0100 MOV AX,0201
3356:0103 MOV BX,0200
3356:0106 MOV CX,0001
3356:0109 MOV DX,0080
3356:010C INT 13          ;读硬盘主引导扇区
3356:010E MOV AX,0301
3356:0111 MOV BX,0200
3356:0114 MOV CX,0008
3356:0117 MOV DX,0000
3356:011A INT 13          ;写软盘8扇区
3356:011C INT 20
3356:011E
-G
-Q
```

#### 程序二 将写入软盘上的硬盘主引导程序写入硬盘第1扇区

```
A)DEBUG
-A 100
3356:0100 MOV AX,0201
3356:0103 MOV BX,0200
3356:0106 MOV CX,0008
3356:0109 MOV DX,0000
3356:010C INT 13          ;读硬盘8扇区
3356:010E MOV AX,0301
3356:0111 MOV BX,0200
3356:0114 MOV CX,0001
3356:0117 MOV DX,0080
3356:011A INT 13          ;写硬盘主引导扇区
3356:011C INT 20
3356:011E
-G
-Q
```

#### 50. PC-DOS 3.30 硬盘分区法与以前的版本有何不同?

PC-DOS 3.30 的命令较之以前的版本新增加 DOS 区内分区功能,即建立 DOS 基本区和扩展区,但有别于供其他操作系统的“非 DOS 区”。DOS 3.30 的硬盘分区命令 FDISK 选择菜单可以选择以下五个方面的功能之一:

1. 建立 DOS 分区
2. 改变活动 DOS 分区
3. 删除 DOS 分区
4. 显示分区信息
5. 选择另一个硬盘驱动器

如果需要在另一硬盘驱动器上使用 FDISK 是选择5。

DOS 3.30 版的 FDISK 建立的 DOS 基本区 C 盘的上限为32MB,扩展区最多可建23个逻辑盘(标识 D-Z),其总容量达726MB。以长城286BH 机配置的 HH-1050 40MB 硬盘为例,可分为10M 的基本区及30MB 的扩展区。这种分法下的 C 盘能接受 DOS2.0 或 2.10 版本的访问,较之30MB 及10MB 顺序为佳。这样为某些系统软件象 CC-DOS 2.13 在大容量硬盘上的使用提供了方便。

#### 51. 怎样用 DM 软件修复硬盘0 磁道“假”坏?

一台 IBM PC/XT286 微机不能从硬盘启动。也不接受 FORMAT 进行格式化,用低级格式化程序例如 LOWFMT.EXE 也能通过,但当进行 FDISK 之后,再用 FORMAT C:/S 进行格式化时却进不去,最后出现“0磁道坏,格式化失败”的错误信息。这时候最好先不要怀疑是硬件问题,用 DM 软件。DM (DISK MANGER)磁盘管理软件具有硬盘物理格式化、硬盘分区、0磁道移道、改变系统配置和检测介质状态等多种功能,是处理硬盘软故障常用的软件。使用时要特别注意屏幕显示信息。

操作步骤如下:

(1)运行 DM 程序,键入 DM 表示选择缺省分区;若键入 DM/M 表示人工定义分区。

A)DM

(2)在主菜单下选“1”,进入物理格式化菜单:

Initialization Menu

Select an option(R):1

(3)如保留缺省分区,键入“Y”后按回车:

Is the above DEFECT-LIST accurate for this drive?(y/n):Y



(4)键入硬盘编号  
 Select an option(R):C  
 Enter Interleave Value(3):1  
 (5)键入“Y”后回车,开始进行物理格式化:  
 THIS WILL DESTROY ANY EXISTING DATA ON THIS DRIVE!  
 Continue?(y/n):y  
 (6)物理格式化结束后,按“R”退回到主菜单,选择“P”,进行预置分区设置:  
 (P)artitioning menu,  
 Select an option (R):P  
 (7)核对分区表,如不修改,键入“Y”后按回车:  
 Does the above PARTITION TABLE regular modification?(y/n)Y  
 (8)预置分区菜单键入“P”后,根据显示信息进行确认:  
 Prepare ALL DOS, Write/Read-only partition?(y/n):Y  
 THIS WILL DESTROY ANY EXISTING DATA ON THIS DRIVE!  
 CONTINUE?(Y/N):Y  
 (9)如在分区内只放置一个系统,键入 Y 后回车:  
 Place a system on the partition?(Y/N):Y  
 (10)写入卷标号  
 Enter 1-11 character Volume Label(RETURN for none):×××

(11)分区设置结束后,先插入系统盘传送系统,再逐级退回 DM,如能从硬盘启动,则故障排除。

在此,也顺便说一下,286档次微机例如长城 286BH、EX 等大都配置40MB 硬盘,DM 软件可以将该容量的硬盘为 C 盘32MB,D 盘8MD,如果出现在 D 盘上安装某个系统,例如 CC-DOS 2.13G 后,逻辑盘 D:“丢失”,显示“无效的驱动器”。此时要求在 C 盘根目录下建立两项条件:

(1)在 CONFIG. SYS 中添加 DEVICE = DM-DRVR. BIN.

(2)拷贝 DMDVR. BIN

两者缺一不可,缺前项屏幕无提示,而后者则显示该文件不存在或者无效。

## 52. 怎样解决从硬盘启动后,屏幕显示:Missing Operating System 错误信息后死机?

硬盘测试成功后便进入 ROM BIOS 的 INT 19 自举程序,把硬盘上0头0柱1扇区的主引导记录和分区信息表读入内存,若检查读入的分区表无引导标志则进入 ROM BASIC,若分区信息表中活动分区多于1个,则显示“无效分区”从而进入死循环。若活动分区唯一,则把活动分区的第一扇区内容读入0000:7C00H 处。若读入错则显示“系统装入错误”进入死循环。若读入正确,则检查读入内容的最后两个字节是否为 AA55,若为 AA55,则把控制权交给活动分区的操作系统,否则显示“Missing Operating System”错误信息而死机。

此时,可将正常 DOS 系统盘上的引导扇区写入硬盘 PC-DOS 分区引导扇区中。操作如下:

A>DEBUG ;进入 DEBUG  
 —L 100 1 0 1 ;DOS 系统盘插入 B 驱动器,读引导扇区  
 —W 100 2 0 1 ;写硬盘 PC-DOS 引导扇区  
 —Q ;退出 DEBUG

## 53. 高密度软盘向普通360KB 软盘复制文件较简单的方法是什么?

CPU 为286、386的中高档微机系统均配置了两个软盘驱动器,一个高密度,一个360KB。例如 GW286BH,其中 A 驱动器为高密度1.2MB 软盘驱动器。高密软盘文件传送到360K 软盘上比较简单的方法是间接拷贝法。这种方法是在硬盘上建立一个子目录,将高密度软盘上的文件拷贝在某一子目录下,然后进入该子目录利用 RESTORE 命令将文件备份到360K 软盘上。但这种方法的缺点是由于在硬盘上频繁地进行拷贝,删除等操作,造成硬盘上的碎片大量增加,使磁头在硬盘上的读写时间延长,从而加速磁头的磨损。

## 54. 怎样在同一个硬盘上安装两个不同的汉字系统?

到目前为止已推出了各种不同版本的汉字操作系统,CC-DOS 2.0/2.10,CC-DOS 2.13A—H,CC-DOS 4.0,GWBIOS 3.00等等,各具特色,在各种机型上广泛使用着。为了发挥出某一汉字系统特殊作用,希望在同一个硬盘上能够存放两个不同的汉字操作系统,实现这一想法的基本思路是将 C 盘分为两个分区,分别安装一种汉字操作系统,使用时采取改变活动分区和分区的系统类型的办法来达到两个不同操作系统共存的目的。

例如在长城0520DH 机20MB 硬盘安装 CC-DOS 4.0和 GWBIOS 3.00,操作步骤如下:

1. 启动 PC-DOS 3.20,运行该系统下的硬盘分区命令 FDISK 删除 DOS 分区,并重新建立 DOS 分区。将101—611柱面的硬盘空间建立为 GWBIOS 3.00 使用的 DOS 分区。这时可在主菜单下选择1,当询问是否将硬盘全部用于 DOS 时回答 N,询问柱面数的开始柱面号时回答510T 101,然后按“ESC”退出 FDISK。

2. 用 PC-DOS 3.20版的 FORMAT/S 命令格式化硬盘,按照 GWBIOS 3.00的安装方法,在硬盘上安装该操作系统。具体操作可分为如下几步:

(1)C>COPY A:\*. \* C:拷贝 PC-DOS 3.20 到硬盘

(2)C>COPY A:\*. \* C:拷贝 GWBIOS 3.00到硬盘

(3)C>RESTORE A: C:将 CLIB24字库文件恢复到硬盘

3. 用 PC-DOS 2.0/2.10系统盘重新启动,运行该版本的 FDISK。由于该版本不能访问 PC-DOS 3.20,因此此时 FDISK 将 PC-DOS 3.20版本 FDISK 建立的系统分区视为 NON-DOS 分区,这样就可以很

方便地将剩余硬盘空间全部或者部分建立 DOS 2.0/2.10 的 DOS 分区。操作方法仍是在主菜单下选择建立 DOS 分区功能并回答柱面数和开始柱面号,若将所有剩余空间用于该 DOS 分区,两次回答均按回车键。然后按“ESC”退出 FDISK。

注意做这一步时,对有的微机可能会出现 DOS 2.0/2.10 版的 FDISK 并不视第一分区为 NON-DOS,这时可使用 PC-DOS 3.20 系统提供的 PARTED.EXE 将第一个分区即 GWBIOS 的分区,改变为 NON-DOS,具体做法是运行 PARTED,在 Menu 中选择 TYPE.,再选(1),然后选 Compatible DOS,最后退出 PARTED 程序即可。

4. 用 PC-DOS2.0/2.10 的 FORMAT/S 命令格式化硬盘该 DOS 分区,然后分别将 CC-DOS 4.0 的 5 张软盘拷贝或恢复到硬盘上。至此,在硬盘上存放 GWBIOS 3.00 和 CC-DOS 4.0。如果这时用硬盘启动,将立即进入 CC-DOS 4.0 汉字系统。

5. 当要改变操作系统,从硬盘启动进入 GWBIOS 3.00 汉字系统,不能用各版本的 FDISK 改变活动分区的办法,因为它不能改变分区系统类型。上面提到的 PC-DOS 3.20 提供的 PARTED.EXE,它具有改变系统类型等很多功能。此时将该程序分别拷贝到硬盘的两个分区中,在 GWBIOS 3.00 和 CC-DOS 4.0 两种汉字系统中均可用它来改变系统类型和活动分区。这样可在任一汉字系统下运行 PARTED.EXE 程序,根据菜单提示选择 TYPE,再选(1)或(2)然后选 Bootable DOS,即将选择的分区类型改为 DOS 系统了,同时将该分区置成活动分区,把另一分区的系统类型改为 NON-DOS,再选择 Quit,退出 PARTED,重新启动系统即可进入所希望的汉字系统了。

在这里顺便说一下,新一代长城系统随机 PC-DOS 3.20 系统盘(1)中有文件:

PFC.BAT  
PARTED.EXE  
DISHZ.COM  
HARDRIVE.SYS

这几个文件可以将大于 33 兆的硬盘作为“D”盘进行格式化并提供用户使用。使用方法如下:

(1) 将 PC-DOS 3.20 插入 A:,键入命令“PFC”  
(2) 在命令 Menu 中选择 CREATE 及 Compatible DOS

(3) 在命令 Menu 中选择 FORMAT 及(2)  
(4) 在命令中选择 QUIT 退出 PARTED.EXE  
(5) 在 C 盘(启动盘)CONFIG.SYS 文件中增加语句:

DEVICE=HARDRIVE.SYS

(6) 重新启动系统。

55. 在高密软盘驱动器中如何格式化 360KB 的软盘?

在高密软盘驱动器中格式化 360KB 软盘只要在 FORMAT 命令中带参数“/4”即可完成,不一定非放

在 360KB 软盘驱动器中进行。命令格式与操作的显示信息如下:

C>FORMAT A:/4

Insert new diskette for drive A:  
and strike ENTER when ready

Format Complete

362496 bytes total disk space

362496 bytes available on disk

Format another? (Y/N)

上述信息清楚地说明在高密软盘驱动器可以格式化 360K 软盘。

56. 40MB 硬盘为什么格式化后变为 32MB 了?

长城 286BH、286EX、AST286 等许多 286、386 的参数执行系统配置 40MB 硬盘,但格式化后只有 32MB,少了 8MB。这是由于 PC-DOS 系统基本 DOS 分区只能达到 32MB。换言之,在对硬盘进行格式化时,只建立了基本 DOS 分区(Primary DOS Partition),大于 32MB 字节硬盘就要建立扩展 DOS 分区(Extended DOS Partition)否则硬盘 DOS 基本分区以外的硬盘空间将造成浪费。

建立扩展 DOS 分区可用硬盘分区命令 FDISK,首先建立基本 DOS 分区,可以指定开始柱面数和柱面数,然后可建立扩展 DOS 分区,也分别给出。但有一点应注意,各分区的柱面数之和一定不能大于硬盘柱面总数。分区以后,返回 FDISK 对 C 盘和 D 盘分解进行 DOS 格式化。

利用磁盘管理软件更容易实现上述操作。

## 培 训 消 息

电子工业出版社电子与电脑杂志将于 93 年 2 月 20 日在北京举办为期一学期的微机硬件维修技术代培班;于 93 年 3 月 31 日举办为期四周的微机硬件二级维修技术学习班。授课内容包括:8088、80286 主机板和 80386、80486 简介;3.5”、5.25”软驱(高低密度)及软卡;硬驱硬卡;CGA、EGA 显示器及 VGA 简介;打印机及打印卡;主机电源;主键盘;小型不间断电源(UPS)等工作原理及实用维修技术,并通过演示课练习故障排除,代培学员除学习上述内容外还将在老师指导下参加各部分实际的维修工作。

代培费:1300.00 元(名额限 4—5 人)

学 费:410.00 元

资料费约 120.00 元(实报实销)

上课、实习、食宿(费用自理)均在中国人民大学校园内。

联系人:北京中国人民大学信息中心

胡野红

电话:255.5431 转 3305 邮编:100872

## C—DBAG 中文数据库应用生成器

F/D \* AG V5.0 零售价:3900元

使用本系统只需按照提示输入用户需求,不需编程,各种管理软件,在功能变动时可任意改动,适用于网络及单用户环境。

功能:生成 DBASE+FoxBASE+源程序;生成任意格式报表;自动生成数据字典及文档;工程图形生成与管理。

组成:下拉/弹出式菜单生成器;查询模块生成器;数据录入与维护模块生成器;数据库文件结构生成器;彩色统计图形模块生成器;任意格式报表生成器;数据字典与文档生成器;

**灵科汉卡 零售价:1200元**

西文软件无需汉化即可直接使用中文,直接写屏,支持1024×768,640×480,640×350图形方式。支持中文排版功能。中文简体与繁体,中文与西文之间一键转换。支持针式打印机、激光打印机及鼠标。精密宋、楷、黑、仿宋体平滑放大输出,能识别中西文窗口。显示字库不占用内存,且显示速度大大提高。

**灵科加密器 零售价:180元**

可加密任何 EXE 和 COM 文件,也可对任何源程序实施加密,如 dBASE, FoxBASE 等,加密文件个数不限。安装方便,只要外插于打印机接口上,且对打印机工作无丝毫影响,加密手段方便,保密性极高。具有强有力的反跟踪功能,使解密成为梦想。

**灵科加密卡 零售价:450元**

具有灵科加密器的一切功能,并且做到了真正的软硬件加密结合的功能。具有开机口令功能,并且可以方便地修改口令或关断开机口令功能。彻底避免了计算机的非授权使用,减少了计算机病毒的感染机会。本卡有2K字节可供编程人员进行二次开发。不用任何其它硬件设备,只需本公司提供的软件工具或函数即可完成向卡内读写加密数据或程序,写入次数以上五万次。本卡极适合保密性高且成本较高软件加密或大、中型 MIS 系统加密使用。

**灵科 MIS 管理新工具**

**程序结构管理及菜单生成系统 零售价:500元**

辅助生成结构图,程序文档;自动生成 WINDOWS、C 语言、dBASE 菜单源程序;支持鼠标操作,模拟显示。

**程序流程图管理系统 零售价500元**

支持顺序判断(IF..ELSE)、循环(WHILE)、选择(SWITCH)逻辑结构图及相应文档;自动生成 C 语言

逻辑调用程序;支持鼠标操作;支持6层嵌套。

经 销:中国电子器材华北公司

地 址:北京市海淀区万寿路西街五号

通信地址:北京144信箱

邮政编码:100036

联系人:石立军 魏 国

电 话:81.1810 81.0920

本公司经营计算机及外设通信设备,欢迎来电函联系。

## 个人电脑联谊活动

为了帮助中小学及个人电脑用户学好用好计算机,解决用户对软件“买难、用难、交流难”的矛盾,星星电脑部在《电子与电脑》编辑部的支持下,开展中华学习机、苹果机、PC 机用户及爱好者联谊活动,欲参加联谊者将登记表填写寄至长春市岭东路115号,星星电脑咨询服务部,随信附手续费、回邮资共1.20元,复函回寄简章、联谊证、软件目录等。

### 主要内容:

1. 相互交换软件、资料;
2. 代销成员自开发的实用软件;
3. 代转让旧机器;
4. 免费咨询相关信息和技术问题;
5. 优惠提供硬件、软盘、资料;
6. 联谊成员交纳30元活动费可成为中华、苹果机沙龙成员,沙龙成员可享受免费拷贝沙龙现有的上千种软件等待遇,详见91年《电子与电脑》第一期。
7. 赠阅电脑信息文摘《家庭电脑》一期。

### 活动形式:

1. 外地会员通过邮寄进行。
2. 本地会员定期举办联谊会,地点设在市少年宫。  
联络部地址:长春市岭东路115号星星电脑部  
邮 编:130031  
联系人:董韶华  
开户行:吉林路城市信用社  
账 号:11029424  
电 话:42002 42727 47823(住宅)  
来访路线:乘61路或1路车终点下南走,岭东商场对面。

# 《电子与电脑》1992年总目录

## ·综述·

| 期                         | 页                |
|---------------------------|------------------|
| 猴年进步——本刊92年致读者 .....      | 1 2              |
| 我国单片机应用技术发展趋势及展望 .....    | 何立民 1 2          |
| 我国单片机应用技术发展趋势及展望(续) ..... | 何立民 2 2          |
| 软件汉化原理 .....              | 郑茂松 3 2          |
| 引人注目的微型机 .....            | 戴伟高 4 2          |
| 91年度 COMDEX 电脑博览会 .....   | 吕问黎 5 2          |
| 美国当今图像处理软件 .....          | 吕问黎 6 2          |
| 优越的专家系统语言 Prolog .....    | 王粤宁 7 2          |
| 计算机信息的安全保护 .....          | 李 群 8 2          |
| 量子芯片与人工智能计算机 .....        | 姚立新 8 4          |
| 计算机安全性的几种实现方法 .....       | 邱向群 9 2          |
| 改革开放结硕果 .....             | 齐 心 10 2         |
| 软件市场大有希望 .....            | 温有良 10 2         |
| 新兴的教育技术—CAI .....         | 陈 健 10 3         |
| 图象处理的一大革命——压缩 .....       | 11 2             |
| 美国消费电子业 .....             | 吕问黎 12 2         |
| 应用计算机网络辅助高中物理教学 .....     | 肖 美 王天谟 李燕萍 12 3 |

## ·PC 用户·

|                                     |          |
|-------------------------------------|----------|
| 自动调平四舍五入误差简易快速软件 .....              | 李燕妹 1 4  |
| 新颖的用户界面一下拉式菜单的实现 .....              | 李晓峰 1 6  |
| 再谈硬盘无效时回收文件的方法 .....                | 岳千钧 1 9  |
| IBM 计算机几个重要硬件设备的软锁程序 .....          | 温盛文 1 10 |
| 一个实用月历程序 .....                      | 沈玉江 1 11 |
| 用拉幕方式显示信息 .....                     | 刘善平 1 11 |
| 拉图游戏 .....                          | 刘晓峰 1 13 |
| 通用文件病毒防御系统 .....                    | 姜金友 1 14 |
| 彩色汉字通用下拉式窗口菜单的设计与实现 .....           | 王 宏 2 4  |
| dBASE III 或 Foxbase 菜单式自由组合查询 ..... | 毛维平 2 9  |
| 单显 PC 机的应用技巧 .....                  | 袁津生 2 10 |
| BASIC 程序用 P 存盘的解密方法 .....           | 杨协贤 2 11 |
| 谈谈编译 BASIC 在长城机上内存不够的问题 .....       | 卢祥江 2 11 |
| 也谈解决运行 FoxBASE 内存不够的问题 .....        | 周立宇 2 12 |

## 编制能灵活打印二维报表的“报表生成系统”

|                              |               |   |    |
|------------------------------|---------------|---|----|
| .....                        | 苏士俊           | 3 | 3  |
| 在管理信息系统中实现计算模型的规则库方法         | ..... 孙永芳 吴泉源 | 3 | 6  |
| dBASE Ⅲ 修改文件属性的方法            | ..... 许再由     | 3 | 8  |
| 抗病毒软件                        | .....         |   |    |
| ——Turbl Anti-Virus V6.80A    | ..... 唐银红     | 3 | 8  |
| 新颖的年历打印通用程序                  | ..... 何贤敏     | 3 | 8  |
| 用24针打印机打印 CAD 图形             | ..... 卢耀志     | 4 | 4  |
| 编制菜单程序的辅助程序@.COM             | ..... 张 红     | 4 | 7  |
| 格式文件与 SET FILTER TO 命令的结合    | .....         |   |    |
| .....                        | 麦红波           | 4 | 8  |
| Azusa/2708病毒与介质描述            | ..... 何悦荣     | 4 | 9  |
| MS FORTRAN 程序的菜单设计技术         | .....         |   |    |
| .....                        | 曹国钧 王 健       | 4 | 10 |
| 如何制做病毒标本                     | ..... 梁宇翀     | 4 | 10 |
| 用 dBASE Ⅲ MEMO 字段开发题库系统      | .....         |   |    |
| .....                        | 陈卫平           | 4 | 11 |
| 关于《dBASE Ⅲ 过程文件加密的发现》质疑      | .....         |   |    |
| .....                        | 张维仁           | 4 | 11 |
| 在单台驱动器上如何作软盘备份               | ..... 杨栋林     | 4 | 12 |
| 一种不打印数值零的方法                  | ..... 曹和平     | 4 | 12 |
| 交互式的 DEL 命令                  | ..... 莫 琳     | 4 | 12 |
| 获取 FOX 文件中密码的两种简单方法          | ..... 周日初     | 5 | 3  |
| 对全屏编辑命令 BROWSE 做一点补充         | .....         |   |    |
| .....                        | 宋开胜           | 5 | 4  |
| 感染 OMICROX 病毒文件的消毒方法         | ..... 刘东祥     | 5 | 5  |
| COLor400卡屏幕彩色图形的硬拷贝          | .....         |   |    |
| .....                        | 张 辉 李东升       | 5 | 6  |
| 利用格式化功能使磁盘“加锁”               | ..... 张李文     | 5 | 9  |
| 一种有效软件解密方法                   | ..... 张彦洪     | 5 | 12 |
| 三维图形显示的遮掩技术及实现               | ..... 季军杰     | 5 | 13 |
| BASIC 超长随机文件的处理方法            | ..... 朱建平     | 5 | 15 |
| dBASE Ⅲ 和 FoxBASE+ 环境下实用多    | .....         |   |    |
| 窗口弹出式菜单的实现方法                 | ..... 张志远     | 6 | 3  |
| CCDOS2.1特殊显示功能的妙用            | ..... 李林枫     | 6 | 4  |
| 在 DOS 中使用功能菜单的方法             | ..... 金林樵     | 6 | 5  |
| PKZIP 压缩软件包的技术背景及使用          | ..... 夏伟文     | 6 | 6  |
| 如何使用《IDOSV1.0》支持 AT101/102键盘 | .....         |   |    |
| .....                        | 杨 旭           | 6 | 9  |
| 恢复硬盘启动功能的方法                  | ..... 王晓红     | 6 | 9  |
| 结构文件与 FIELD 函数的比较            | .....         |   |    |
| .....                        | 严桂兰 刘甲耀       | 7 | 4  |





|                                   |           |    |
|-----------------------------------|-----------|----|
| 二维函数曲线拟合 .....                    | 李 涛 8     | 17 |
| 在 Applesoft 系统下实现 COMMON 功能 ..... | 安赵根 8     | 18 |
| 通用打印课表程序 .....                    | 程建伟 8     | 19 |
| CP/M 系统文件的恢复方法 .....              | 李 齐 8     | 20 |
| 英语单词快速记忆系统 .....                  | 张振堂 9     | 13 |
| 用 CHR\$ 函数压缩存盘数据 .....            | 梁才柱 9     | 16 |
| PB-700 微机解密技巧 .....               | 孙 力 9     | 17 |
| 双人百米赛跑 .....                      | 汪 波 9     | 17 |
| 寻找莱蒙托夫游戏规律 .....                  | 钱雁群 9     | 17 |
| 也谈“一题多解” .....                    | 郑明达 9     | 18 |
| 考试的统计分析程序 .....                   | 胡筱罡 9     | 19 |
| 氯化氢制取的动态显示 .....                  | 刘 萍 9     | 20 |
| 磁盘加密一法 .....                      | 黄晓晖 9     | 21 |
| BASIC 子程序的递归调用 .....              | 陈继良 丘 文10 | 15 |
| APPLE 机的快速排序 .....                | 张 亭10     | 16 |
| 百年公历—农历互查 .....                   | 刘安军19     | 17 |
| 内存数据代码搜索程序 .....                  | 苏 华10     | 20 |
| 微机模拟游标卡尺读数训练程序 .....              | 王太文11     | 17 |
| 磁带虚拟磁盘的文件管理 .....                 | 胡发新11     | 18 |
| APPLE— I 音乐功能扩充 .....             | 陈建明11     | 19 |
| POSITION 命令新用 .....               | 周 进11     | 20 |
| CEC—I 与1724打印机配接图形硬拷贝程序 .....     | 张益贵11     | 20 |
| 游戏接口 PDL 的扩展 .....                | 张建群11     | 21 |
| SUPER DOS 简介 .....                | 张 志11     | 21 |
| 推荐一个扩展 BASIC 语句——拆字串语句 .....      | 王 凯12     | 20 |
| 也谈内存信息的打印 .....                   | 张 亭12     | 22 |
| 增强中华学习机的音响功能 .....                | 杨云霄12     | 23 |
| 计算机各类债券储蓄收益利息 BASIC 程序 .....      | 陶文庆12     | 24 |
| 哥德巴赫猜想 BASIC 程序的改进 .....          | 卢良红12     | 24 |
| 数组的动态删除 .....                     | 冯端品12     | 25 |

## 520 机器语言讲座·

|                          |        |    |
|--------------------------|--------|----|
| 6520 机器语言程序设计讲座 .....    | 朱国江 1  | 23 |
| 第二章 微处理器、存储器、寄存器简介 ..... | 朱国江 2  | 20 |
| 第三章 6520 MPU 的寻址方式 ..... | 朱国江 3  | 22 |
| 第四章 6520 MPU 的指令系统 ..... | 朱国江 4  | 22 |
| 第五章 源程序的编辑、汇编和运行 .....   | 朱国江 5  | 24 |
| 第六章 简单程序和分支程序设计 .....    | 朱国江 6  | 23 |
| 第七章 循环程序设计 .....         | 朱国江 7  | 20 |
| 第八章 子程序设计 .....          | 朱国江 8  | 21 |
| 第九章 堆栈程序设计 .....         | 朱国江 9  | 21 |
| 第十章 监控子程序的调用 .....       | 朱国江 10 | 21 |
| 第十一章 监控子程序的调用(下) .....   | 朱国江 11 | 23 |

## ·FORTH 语言讲座·

|              |       |    |
|--------------|-------|----|
| 第一讲 概述 ..... | 丁志伟12 | 26 |
|--------------|-------|----|

## ·初级程序员级水平考试辅导讲座·

|                 |       |    |
|-----------------|-------|----|
| 计算机硬件基础知识 ..... | 顾育麒 1 | 27 |
|-----------------|-------|----|

|                          |       |    |
|--------------------------|-------|----|
| 计算机硬件基础知识(续) .....       | 顾育麒 2 | 24 |
| 第三章 数据结构 .....           | 宋丹颖 3 | 26 |
| 操作系统及 PC-DOS .....       | 宋丹颖 4 | 27 |
| 数据库基础 .....              | 阚 路 5 | 29 |
| BASIC 基础 .....           | 李志刚 6 | 29 |
| PC BASIC 自测试题解答与分析 ..... | 李志刚 7 | 25 |
| 一九九二年计算机初级软件人员竞赛试题 ..... | 8     | 26 |
| 试题解答与分析 .....            | 9     | 25 |

## ·学用单片机·

|                                     |           |    |
|-------------------------------------|-----------|----|
| 8031 单片机最小系统 .....                  | 罗明宽 李金相 1 | 31 |
| 低功耗单片机最小系统 .....                    | 张培仁 刘振安 2 | 29 |
| 新一代超高集成度 Z80 微处理器系列 .....           | 董伯明 2     | 31 |
| 单片机最小应用系统与液晶显示器(LCD)的接口 .....       | 张培仁 3     | 29 |
| 实现单片机最小系统与 PC 机的通信 .....            | 张培仁 4     | 30 |
| 提高单片机最小系统抗干扰能力和自恢复方法 .....          | 张培仁 刘振安 5 | 32 |
| TP801 单板机与微型机数据双向并行传送一例 .....       | 刘得和 5     | 33 |
| MCS—51 汇编指令外延的应用 .....              | 陈亿善 5     | 34 |
| 带 A/D 的单片机最小应用系统 .....              | 张培仁 刘振安 6 | 31 |
| BJS—51 单片机实验系统 .....                | 盛焕鸣 7     | 29 |
| BJS—51 的监控程序 .....                  | 盛焕鸣 8     | 31 |
| 单片机实验与 BJS—51 实验教程 .....            | 李广弟 9     | 30 |
| BJS—51 单片机实验系统(续) .....             | 李广弟 10    | 25 |
| CYSCB—ZMCS—518098 单片单板机硬件设计原理 ..... | 吴 微 10    | 26 |
| BJS—51 单片机实验系统(续) .....             | 张俊漠 11    | 26 |
| 模/数、数/模转换实验(续) .....                | 张俊漠 12    | 29 |

## ·学装微电脑·

|                        |        |    |
|------------------------|--------|----|
| 附带应急开关的顺序控制 .....      | 易齐干 1  | 24 |
| 打印数字温度计 .....          | 易齐干 2  | 32 |
| 电源 ON、OFF 自动运转装置 ..... | 易齐干 3  | 32 |
| 红外线遥控室内温度 .....        | 易齐干 4  | 34 |
| 制作能暂停显示的计数器 .....      | 易齐干 5  | 35 |
| 7 段数码管读数装置 .....       | 易齐干 6  | 34 |
| 步进电机的控制 .....          | 易齐干 7  | 33 |
| 水平多关节型机器人 .....        | 易齐干 8  | 34 |
| 自动输送装置 .....           | 易齐干 9  | 32 |
| 微电脑控制微型钻床 .....        | 易齐干 10 | 30 |
| 微电脑控制微型钻床 .....        | 易齐干 11 | 29 |

## ·电脑巧开发·

|                                     |           |    |
|-------------------------------------|-----------|----|
| 多功能程序移植器——一种简易的硬件工具 .....           | 朱立纲 1     | 37 |
| 单板机 RAM 分段与程序保护的实现 .....            | 王新明 2     | 37 |
| 微机视频接口器的研制 .....                    | 韦江维 覃龙生 3 | 37 |
| XMF 学习机(单驱扩展箱)改为 XMF 及 PC 两用机 ..... | 杨青海 3     | 38 |

|                                 |         |    |    |                                   |                 |    |    |
|---------------------------------|---------|----|----|-----------------------------------|-----------------|----|----|
| Z80单板机与打印机巧相连 .....             | 张慧成     | 4  | 37 | TH3070点阵式打印机常见故障分析与检修(下)          | 刘立华             | 6  | 44 |
| 降低 MP-1 电脑的功耗 .....             | 邓文超     | 4  | 40 | Super AT 机显示电源原理及维修 .....         | 范志盛             | 7  | 45 |
| 简易字幕灯的制作与控制 .....               | 杨亮泽     | 5  | 38 | 利用 PCTOOLS 校正软盘驱动器磁头 .....        | 黄焕如             | 7  | 48 |
| DOS 系统下多种软件在一台微机上的共存方法 .....    | 尤建忠     | 5  | 40 | 单色显示器故障维修一例 .....                 | 梁建华             | 8  | 43 |
| 自带 EPROM 类单片机简易加密编程 .....       | 庄凯      | 6  | 38 | 恢复 CEC-I 学习机 DOS 的 I/O 控制 .....   | 屈晓柳             | 8  | 43 |
| 一种简单的键盘接口电路 .....               | 罗许建     | 6  | 39 | 再谈清洗盘的正确使用 .....                  | 刘德一             | 8  | 44 |
| ASCII 码字符显示器设计 .....            | 李德文     | 7  | 38 | 对磁盘局部缺损的处理 .....                  | 何管略             | 8  | 44 |
| 点阵字符液晶显示器与单片机接口 .....           | 张培仁 杨建景 | 8  | 38 | IBM-PC/XT 及其兼容机 RAM 故障的检修方法 ..... | 齐吉泰             | 9  | 39 |
| 计算机语音输出功能的开发与应用(上) .....        | 陈竹林     | 9  | 36 | APPLE-I CEC-I 故障诊断仿真系统 .....      | 王志刚 张一建         | 10 | 43 |
| 怎样使 CCDOS2.10 的九针打印驱动程序适应 ..... | 李修连     | 9  | 38 | 笔记本型电脑技术和市场 .....                 | 赵广恩             | 11 | 43 |
| 2.13H 汉字系统 .....                | 肖革文     | 9  | 38 | 用软件对软驱进行简单读写检测 .....              | 李晓中 麻佳洛 人府静 张景生 | 11 | 44 |
| 8031 真的无 ROM 吗? .....           | 陈竹林     | 10 | 35 | LQ-1600K 打印机接口专用集成电路 .....        | 赵继文             | 12 | 38 |
| 计算机语音输出功能开发与应用(下) .....         | 王正英     | 11 | 35 | M54610P 的简易修复方法 .....             | 杨远成             | 12 | 40 |
| 数字集成电路简易测试器 .....               | 李永和     | 11 | 39 | GW286 计算机硬盘控制卡故障处理一例 .....        |                 |    |    |
| 中华学习机调制伴音电路 .....               | 邓本富     | 12 | 35 |                                   |                 |    |    |
| 中华学习机巧测电容 .....                 | 聂铁轮     | 12 | 33 |                                   |                 |    |    |
| 家庭如何配置中华学习机 .....               |         |    |    |                                   |                 |    |    |

## ·电脑游戏机·

|                          |    |    |    |
|--------------------------|----|----|----|
| 第二章 FBASIC 的基本语句 .....   | 于春 | 1  | 39 |
| 第三章 FBASIC 的画面控制语句 ..... | 于春 | 2  | 41 |
| 第三章 FBASIC 的画面控制语句 ..... | 于春 | 3  | 39 |
| 第三章 FBASIC 的画面控制语句 ..... | 于春 | 4  | 41 |
| 第四章 FBASIC 语言的深入理解 ..... | 于春 | 5  | 41 |
| 第四章 FBASIC 语言的深入理解 ..... | 于春 | 6  | 40 |
| 第四章 FBASIC 语言的深入理解 ..... | 于春 | 7  | 41 |
| FBASIC 语言的游戏程序编写技巧       |    |    |    |
| 第一讲 游戏程序概念 .....         | 于春 | 8  | 41 |
| 第二讲 故事情节和游戏结构 .....      | 于春 | 9  | 41 |
| 第三讲 程序结构和程序框图 .....      | 于春 | 10 | 38 |
| 第四讲 游戏程序的设计过程(上) .....   | 于春 | 11 | 40 |
| 第四讲 游戏程序的设计过程(中) .....   | 于春 | 12 | 35 |

## ·维修经验谈·

|                                |     |   |    |
|--------------------------------|-----|---|----|
| 微机安装和使用中应注意的几个问题 .....         | 胡野红 | 1 | 43 |
| CEC-I 中华学习机修理一例 .....          | 卢光怀 | 1 | 44 |
| 软盘驱动器综合故障维修一例 .....            | 周凯歌 | 1 | 44 |
| PC-1500 计算机的维修 .....           | 何静  | 2 | 44 |
| 导电橡胶按钮被磨损的修理方法 .....           | 梁绍建 | 2 | 44 |
| IBM-PC 机故障维修一例 .....           | 欧阳波 | 2 | 44 |
| NP125 复印机中一个易被忽视的故障 .....      | 许鹰  | 3 | 42 |
| PC-88 键盘分析与故障维修 .....          | 王耀亭 | 3 | 42 |
| APPLE I 电源常见故障维修方法 .....       | 邓满园 | 3 | 44 |
| IBM PC/XT 机的软磁盘驱动器磁头校准程序 ..... | 范思尧 | 3 | 45 |
| 中华学习机电源故障维修经验谈 .....           | 汪晖  | 4 | 44 |
| 袖珍计算机 PC 型 PB 型系统 RAM 扩展模块     |     |   |    |
| 常见故障与维修 .....                  | 朱杰  | 4 | 46 |
| TH3070 打印机常见故障分析与检修(上) .....   | 刘立华 | 5 | 44 |
| 维修打印头应注意的几个问题 .....            | 蔡世清 | 6 | 44 |

## ·新书与软件·

|                          |       |    |
|--------------------------|-------|----|
| 电子工业出版社软件部新出版软件介绍 .....  | 1     | 45 |
| DATA BASE IV 关系数据库 ..... | 张冰毅 2 | 45 |
| 电子工业出版社软件部新出版软件介绍 .....  | 8     | 45 |
| 软件介绍 .....               | 9     | 44 |

## ·读者联谊·

|                                   |     |    |    |
|-----------------------------------|-----|----|----|
| 普及型 PC 个人用户软件交流联谊活动问题解答(一) .....  | 王路敬 | 1  | 46 |
| 普及型 PC 个人用户软件交流联谊活动问题解答(二) .....  | 王路敬 | 2  | 46 |
| 显示器·显示卡·显示系统 .....                | 孙梅英 | 2  | 48 |
| 普及型 PC 个人用户软件交流联谊活动问题解答(三) .....  | 王路敬 | 3  | 46 |
| 普及型 PC 个人用户软件交流联谊活动问题解答(四) .....  | 王路敬 | 4  | 48 |
| 普及型 PC 个人用户软件交流联谊活动问题解答(五) .....  | 王路敬 | 5  | 47 |
| 普及型 PC 个人用户软件交流联谊活动问题解答(六) .....  | 王路敬 | 6  | 48 |
| 普及型 PC 个人用户软件交流联谊活动问题解答(七) .....  | 王路敬 | 7  | 48 |
| 普及型 PC 个人用户软件交流联谊活动问题解答(八) .....  | 王路敬 | 8  | 46 |
| 普及型 PC 个人用户软件交流联谊活动问题解答(九) .....  | 王路敬 | 9  | 46 |
| 普及型 PC 个人用户软件交流联谊活动问题解答(十) .....  | 王路敬 | 10 | 46 |
| 普及型 PC 个人用户软件交流联谊活动问题解答(十一) ..... | 王路敬 | 11 | 46 |
| 普及型 PC 个人用户软件交流联谊活动问题解答(十二) ..... | 王路敬 | 12 | 46 |