

E&C

1992

●一九九二年 ●总期第90期

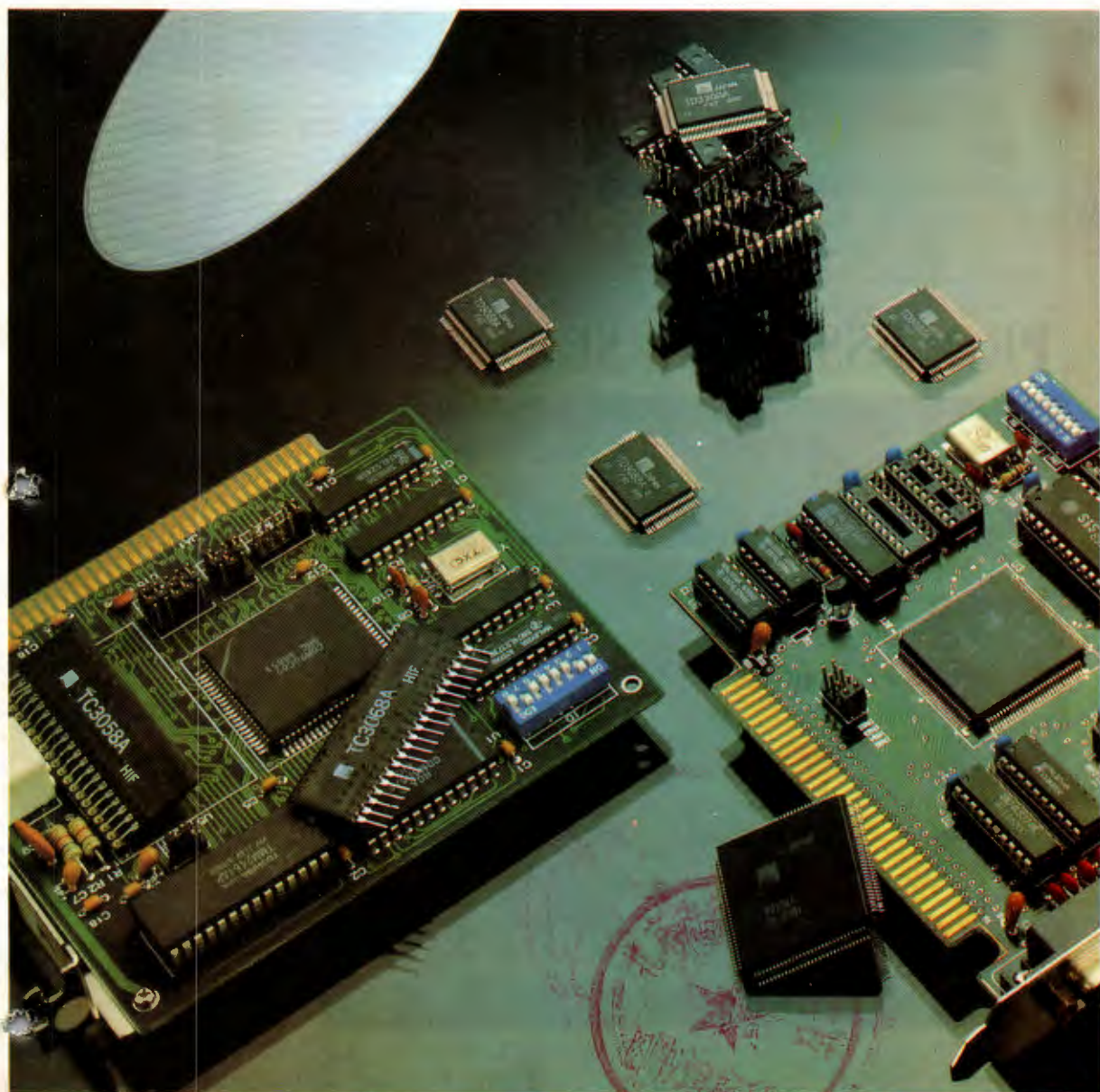
9

電子

ISSN 1000-1077

4/5
479

與 電腦



• **ELECTRONICS AND COMPUTERS** •



武汉创意电子研究所隆重推出

CYSC13-2MCS-51,8098单片单板机

单片机应用文集

何立民 主编

SCB-2型MCS-51, 8098单片单板机

机械工业出版社 总 发 局 国 际 机 械 工 业 出 版 社

● ●

[illegible]

附： 8.2.2 可燃气体的检测与报警

A/D, 外部的A/D是ADC0809, 它是8位的A/D。它是8086单片机, 芯片内还有4路10位A/D。

D/A: 外部配的D/A是DAC0832。它是1路8位D/A。若是8086单片机, 则片内还有1路数模转换输出。

输出：本机配有25个压电式水接嘴，其中一个为复位接嘴，18个为喷射接嘴，8个为全接嘴。

其内部码管组成, 4 个用来显示地址状态信息, 2 个显示数

8086 8 位单片微机硬件 (1)

因此，在材料选择时，应尽量选择与基体材料相容性好的材料。

吴 微 文 军 编著

EKE **RDS**

单片机原理 及制作

武汉大学出版社

国内代号: 2-888

定价：0.95元

XENIX 系统提高培训录相带简介

“XENIX 系统提高培训”录相包括三大部分内容：第一部分是 XENIX 系统的工作原理、第二部分是 XENIX 系统的开发工具，第三部分是 XENIX 系统的维护。它要求听课者应具有“XENIX 入门及使用”知识，针对的是 XENIX 系统的工具软件和应用软件的开发人员以及 XENIX 系统的系统管理员。本录相内容具体分为如下六章：

第一章 XENIX 系统概况

这一章概括地介绍了 XENIX 系统的发展情况，XENIX 系统的特点以及 XENIX 系统上可运行的软件，使听课人员对 XENIX 系统有一个比较概括而又全面的了解。

第二章 XENIX 系统的结构及工作原理

这一章分别介绍了 XENIX 系统的整体结构，XENIX 系统的核心结构和 XENIX 核心的工作原理。在工作原理中，对 XENIX 核心的四个组成部分——文件系统、设备管理、进程管理及内存管理分别进行了深入浅出而又有条理的讲析。这一章旨在使听课人员对 XENIX 系统的内部结构及工作原理有一个较全面而又清晰的了解，这无论对开发人员还是系统管理人员都是必不可少的基础。

第三章 XENIX 系统调用和 C 语言库函数

本章首先讲解了 XENIX 系统调用和 C 语言库函数中所用到的用户常见术语；然后对 XENIX 系统调用进行了一一讲析并举例讲解；最后对 C 语言库函数进行了分类介绍及举例讲析并讲了开发人员如何建立自己的函数库，对 XENIX 开发的工作人员来说，用这些系统调用和库函数已基本能够进行 XENIX 系统下的工具软件和应用软件的开发。

第四章 开发工具

这一章讲了反汇编工具 adb 和大型程序开发、维护工具 make，这无论对 XENIX 系统的开发人员还是系统管理员都是很有用的工具。

第五章 外部设备的连接

这一章以终端（以及多用户卡）、硬盘、磁带机及打印机为例讲了 XENIX 系统外部设备的连接，这对系统管理和维护人员是很有用的。

第六章 系统维护

该章讲了 XENIX 系统管理人员日常应该做的维护工作，并专门讲解了 XENIX 系统的参数配置，也就是资源分配的原理及方法，这一章主要针对系统管理人员。

注：参考教材：

录相带内容的第一章、第二章、第三章及第六章的第二节分别是浪潮培训中心《XENIX SYSTEMV 培训教材》补充教材的第一章（XENIX 系统概况）、第二章（XENIX 系统工作原理）、第三章（XENIX 系统调用）、第四章（XENIX 的 C 语言库函数）及第五章（系统参数配置）。

录相带内容的第四章、第五章及第六章的第一节分别是浪潮培训中心《XENIX SYSTEMV 培训教材》的第十一章、第十四章、第十七章及第十章。

需要者可与济南山大路 224 号，邮编 250013（浪潮电子信息产业集团公司培训中心）何积功同志联系。

庆祝电子工业出版社建社十周年

1982—1992 年

将部分新书、优秀图书奉献给读者

序号	书 名	邮购价	序号	书 名	邮购价
9-1	薄膜物理	13.80	9-27	18 英寸彩色电视机大全	17.00
9-2	针式打印机	8.60	9-28	国产 14、17 英寸黑白电视机维修 图册全集(上、中、下)	47.30/套
9-3	电源	8.60	9-29	中外高保真与组合音响维修手册	15.50
9-4	微型计算机维修实例 999	14.40	9-30	最新进口平面直角彩色电视机维 修手册(一、二)	46.60/套
9-5	MS-DOS3.30STD 系统 BIOS 分析 应用与虚盘管理	11.00	9-31	家电维修技术精华丛书(全国征文 大奖赛丛书 1~10 集)	
9-6	UNIX 系统安全实用技术指南	9.40	9-32	国内外流行摩托车电器设备原理 维修及图集	13.60
9-7	UNIX 操作系统设计与实现	13.40	9-33	实用空调制冷设备维修大全	18.60
9-8	UNIX 使用大全	13.50	9-34	卫星电视接收机原理与制作	11.30
9-9	DOS 使用大全	15.50	9-35	常用电子电器电路精选	11.00
9-10	C 语言大全	14.50	9-36	电视机部件器件元件故障的判断 检测修理代换	13.50
9-11	dBASE III PLUS 大全	15.00	9-37	黄河牌电视机用户指南	18.40
9-12	汉字 dBASE III 原理与应用	6.00	9-38	电视新技术—原理·器件·系统 和设计	22.00
9-13	汉字 dBASE III 实用教程	10.40	9-39	电子测量仪器选购指南	40.30
9-14	汉字 dBASE III 教程	11.00	9-40	555 集成电路应用 800 例	28.80
9-15	怎样使用中西文 FOXBASE+V2. 10	17.00	9-41	日英汉机电技术大词典(上、下)	188.00
9-16	Quick BASIC 程序设计	10.40	9-42	汉俄实用词典	57.50
9-17	Turbo C 语言程序设计	15.60	9-43	实用日语句型	12.70
9-18	AUTO CAD 操作手册(2.6~9.03 版)	18.40	9-44	中外电信集成电路使用手册	25.90
9-19	长城 0520 微型计算机实用教程	9.10	9-45	实用电子器件和电路简明手册	34.50
9-20	PC 机通用软件操作手册(一、二)	20.60/套	9-46	黑白电视机原理与检修	13.40
9-21	现代计算机技术博览(一)	28.80	9-47	彩色电视机原理与维修	8.10
9-22	计算机应用指南	17.30	9-48	收录机原理与维修	9.00
9-23	电子与电脑合订本(91 年)	17.80	9-49	电子线路	8.30
9-24	全国中小学计算机教育资料汇编	15.60	9-50	电子琴原理使用与维修	9.20
9-25	计算机应用基础	16.10	9-51	机械电子工业年鉴(91 年)	46.00
9-26	实用 DOS 详解词典	15.00			

邮购方法·邮局汇款:100036 北京万寿路电子工业出版社邮购科

银行汇款·开户行:北京市工商银行翠微路分理处

帐号:661036—40 请将清单一同随汇单寄来

电话:813693 电挂 3101

一九九二年

总期第90期

電子與電腦

• ELECTRONICS AND COMPUTERS •

目 录

• 综述 •

计算机安全性的几种实现方法 邱向群(2)

• PC 用户 •

文本方式下汉字菜单的实现 曾跃忠 蔡以群(3)

巧解黑色星期五病毒 谭小敏(5)

通用的汉化 WordStar 曹国钧(6)

反函数查找法的实现 王杰民(7)

在 CCDOS2.13F 下支持下 dBASE III 画图技术
..... 徐国茂 黄松德 刘少明(8)

• 学习机之友 •

英语单词快速记忆系统 张振堂(13)

用 CHR\$ 函数压缩存盘数据 梁才柱(16)

PB-700 微机解密技巧 孙 力(17)

双人百米赛跑 汪 波(17)

寻找莱蒙托夫游戏规律 钱雁群(17)

也谈“一题多解” 郑明达(18)

考试的统计分析程序 胡筱翌(19)

氯化氢制取的动态显示 刘 萍(20)

磁盘加密一法 黄晓晖(21)

• 6502 机器语言讲座 •

第九章 堆栈程序设计 朱国江(22)

• 初级程序员级软件水平考试辅导 •

试题解答与分析 (25)

• 学用单片机 •

单片机实验与 BJS-51 实验教程 李广弟(30)

• 学装微电脑 •

自动输送装置 易齐干(32)

• 电脑巧开发 •

计算机语音输出功能的开发与应用(上)
..... 陈竹林(36)

怎样使 CCDOS2.10 的九针打印驱动程序适应

2.13H 汉字系统 李修连(38)

8031 真的无 ROM 吗? 肖革文(38)

• 维修经验谈 •

IBM-PC/XT 及其兼容机 RAM 故障的检修方法

..... 齐吉泰(39)

• 电脑游戏机 •

第二讲 故事情节和游戏结构 于 春(41)

• 新书与软件 •

软件介绍 (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(九)

..... 王路敬(46)

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

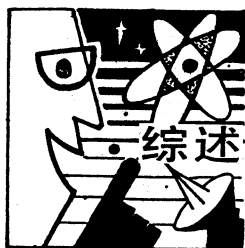
订购处:全国各地邮电局

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



计算机安全性的几种实现方法

西安交通大学(710049) 邱向群



由于近年来计算机病毒的泛滥,各式各样的数据窃取以及计算机犯罪,人们对计算机的安全性越来越重视。

目前普遍使用的保护方法主要是对合法用户的身份进行验证:例如阅读磁卡,指纹检验和密码口令校验等。

根据美国计算机安全委员会近来的一项调查,发现所有计算机损失事件中仅有3%是由外部非法用户造成的,65%的损失是由于雇员的差错而造成的,其余32%是那些不忠诚或为了泄私愤的雇员故意造成的。事实上很多人对计算机安全性的认识不足,不少人将软磁盘和写有密码的本子随意乱放,与别人共用磁盘,选择最易于被猜到的口令,例如生日,一星期内的日期或身份证等证件上的号码。一个单位竟有数十人用单位电话号码做密码。

下面给出一种美国普遍使用的RSA加密算法:这一算法是美国麻省理工学院的三位科研人员研制的。为了便于说明问题,下面的例子使用简单的质数,而在实际运用中,密码通常是一个很大很复杂的数,使非法用户几乎无法解密。

一、生成加密码:

1. 随机地取一个奇数: $E=5$;
2. 取二个质数 P 和 Q , $P=7, Q=17$; 使 $(P-1)(Q-1)-1$ 可以被 E 整除;
3. P 乘 Q 得 N : $P \times Q = N = 7 \times 17 = 119$;
4. 将 N 和 E 连接起来即为加密码 NE : $NE = 1195$ 。

二、生成解密码:

1. P, Q 和 E 分别减1后相乘,再加1:
 $(P-1)(Q-1)(E-1)+1 = 6 \times 16 \times 4 + 1 = 385$;
2. 将结果除以 E , 得到 D : $D = 385 / 5 = 77$;
3. 将 N 和 D 连接起来得到解密码 ND : $ND = 11977$ 。

三、用加密码对数据加密:

1. 先将数据转化成数字,例如用19代表S, 则明码=19。
2. 加密算法:
1) 对明码进行 E 次幂运算: $19^5 = 2476099$;
2) 结果除以 N : $2476099 / 119 = 20807$, 余数为66;
3) 余数即为密码: 密码=66。

四、用解密码对密码解密:

1. 解密算法:
1) 对密码进行 D 次幂运算: $66^{77} = 1.27 \times 10^{140}$;
- 2) 结果除以 N : $1.27 \times 10^{140} / 119 = 1.069 \times 10^{138}$, 余数

为19;

2. 余数即为明码: 明码=19。

有不少专家认为密码法对计算机并非万无一失,他们中的一些人提出并实现了一些改进方法,例如三次口令输入错误(一般合法用户是不会犯这样的错误的)就退出系统,并发出警报,或是再配合硬件一起使用——目前主要是磁卡,有人称此为动态口令,上面的密码是用户无法阅读或修改的。目前美国约有70,000~100,000计算机用户使用这种方法,用户主要是国防部门、航空航天、通信以及金融等要害机构。一般工业和民用用户对目前颇为昂贵的磁卡还不敢问津。

“生物法”是一种高科技的身份校验方法,它通过校验人体的生理特点或行为特性来防止非法用户的侵入。生理特点如指纹、视网膜上的血管布局 and 手的几何形状——这些特点因人而异,而对每个人来说则是固定不变的。行为特点如声音、动态签字方式以及敲击键

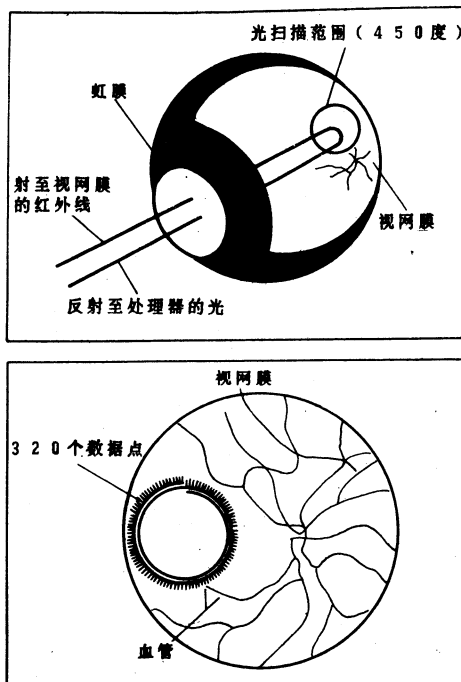


图1

(转第12页)



PC 用户

文本方式下汉字菜单的实现

苏州三光电加工有限公司 曹跃忠 蔡以群

弹出式菜单通过屏幕把用户和系统联系起来,显示直观、选项快速方便,因而深得广大用户青睐,已被越来越多的系统所采用。但是,由于弹出式菜单需要把显示区内容保存在内存中,因此弹出式菜单一般都是在文本方式下以西文显示的方式实现的,这对许多不太熟悉西文的用户来说具有明显的缺陷。

在系统开发的过程中,我们对计算机的显示原理作了一些研究,实现了 EGA/VGA 显示器上文本方式下的汉字显示,从而开发成功了汉字显示的弹出式菜单,并在我们的实际系统中取得了令人满意的效果。

一、EGA/VGA 文本方式下的显示原理

EGA/VGA 显示卡上可以有不同数目的存储器,这些存储器一般分成四个独立的等容量的存储区段,即彩色页面。对于 CPU 来说,这些彩色页面具有相同的存储地址,CPU 执行读或写操作时,彩色页面的选通与否由相关寄存器的设置值来决定。

在文本模式下,彩色页面 0 和彩色页面 1 存放 ASCII 值及其文本属性,彩色页面 2 存放字符集。在彩色页面 1 中存放的文本属性是指相应的显示于屏幕上的 ASCII 字符的显示属性,它具有 8 个规定位,在标准彩色文本模式下,各位含义如表 1 所示:

D ₇	D ₆	D ₅	D ₄	D ₃	D ₂	D ₁	D ₀
----------------	----------------	----------------	----------------	----------------	----------------	----------------	----------------

- D₂、D₁、D₀——背景彩色
 D₃——前景增强或字符表选择
 D₆、D₅、D₄——底色
 D₇——底色增强或闪烁

表 1. 文本属性位

当一个 ASCII 字符要显示在屏幕上时,计算机首先从页面 2 的字符表中读取 ASCII 字符的相应的点阵信息,据此再向显示器输出显示信息。所谓字符表,就是指希望在屏幕上显示的符号的点阵信息的集合。

含有 256KRAM 的 EGA/VGA 显示卡的每个页面存储容量为 64K,每个字符表包含 256 个字符,每个字符占 32 个字节,每个字符表长为 8K,因此页面 2 可以包含多个字符表,表 2 是字符表的地址。

offset		offset	
E000H	未用	E000H	字符表 8
C000H	字符表 4	C000H	字符表 4
A000H	未用	A000H	字符表 7
8000H	字符表 3	8000H	字符表 3

6000H	未用	6000H	字符表 6
4000H	字符表 2	4000H	字符表 2
2000H	未用	2000H	字符表 5
0000H	字符表 1	0000H	字符表 1

EGA

VGA

表 2. RAM 中字符表的驻留地址

由于 RAM 中可以驻留多个字符表,因此可以通过设置定时器中相应中的寄存器来选择 1 个或 2 个活化字符集,字符表选择寄存器各位含义如表 3 所示。

D ₇	D ₆	D ₅	D ₄	D ₃	D ₂	D ₁	D ₀
----------------	----------------	----------------	----------------	----------------	----------------	----------------	----------------

- D₁、D₀——字符表选择 B
 D₃、D₂——字符表选择 A
 D₄——字符表选择 B(对 VGA)
 D₅——字符表选择 A(对 VGA)
 D₇、D₆——保留

表 3. 字符表选择寄存器

当用户选择了两个字符表后,文本属性中的位 D₃起选择字符表的作用,D₃=0 表示选择字符表 B,D₃=1 则表示选择字符表 A。

二、文本方式下汉字菜单的实现

从 EGA/VGA 文本方式下的显示原理可以知道,屏幕显示是由字符表中的点阵信息来决定的,因此当我们在字符表中装入汉字点阵信息时,汉字就会显示在屏幕上。

在一般的实际应用系统中,所需要显示的汉字不会很多,因此我们可以通过设置字符表选择寄存器来活化二个字符集,在系统中就可以活化二个字符表,一个字符表存放 ASCII 字符的点阵信息,而另一个字符表则存放汉字点阵信息,这样可以为系统提供一百多个汉字。如果在系统中变换汉字字符表,就可以为系统提供任意多个汉字。

为了装入汉字信息显示的字符表,必须对有关寄存器值进行修改,汉字字符表装入后,又必须对相关寄存器进行信息恢复,表 4 提供了几个我们用 Turbo C 编写的函数,其中 setvga() 完成装入汉字字符表前的寄存器值修改,resetvga() 在装入汉字字符表后对寄存器值进行恢复,而 sel_char(t₁, t₂) 则选择字符表, t₁ 表示选择字符表的个数, t₁=1 表示选择 1 个字符表, t₁≠1 则表示选择 2 个字符表, t₂ 则表示选择哪一个字符表。这些函数适用于 EGA 或 VGA。

setvga()

```

char seqparm[] = {0x00, 0x02, 0x04, 0x00};
char seqparm1[] = {0x01, 0x04, 0x07, 0x03};
char gcparm[] = {0x04, 0x05, 0x06};
char gcparm1[] = {0x02, 0x00, 0x00};
int i;
inportb(0x0da);
outportb(0x03c0, 0x10);
outportb(0x03c0, 0x00);
for(i=0; i<=3; i++)
{
    outportb(0x03c4, seqparm[i]);
    outportb(0x03c5, seqparm1[i]);
}
for(i=0; i<=2; i++)
{
    outportb(0x03ce, gcparm[i]);
    outportb(0x03cf, gcparm1[i]);
}
}

```

resetvga()

```

{
    char seqparm[] = {0x00, 0x02, 0x04, 0x00};
    char seqparm1[] = {0x01, 0x03, 0x03, 0x03};
    char gcparm[] = {0x04, 0x05, 0x06};
    char gcparm1[] = {0x00, 0x10, 0x0e};
    int i;
    for(i=0; i<=3; i++)
    {
        outportb(0x03c4, seqparm[i]);
        outportb(0x03c5, seqparm1[i]);
    }
    for(i=0; i<=2; i++)
    {
        outportb(0x03ce, gcparm[i]);
        outportb(0x03cf, gcparm1[i]);
    }
}

```

sel __char(int t1, int t2)

```

{
    union REGS regs;
    if(t1==1)
    {
        regs.x.ax=0x1103;
        regs.h.bl=t2;
        int86(0x10, &regs, &regs);
    }
    else
    {
        regs.x.ax=0x1000;
        regs.x.bx=0x0712;
    }
}

```

```

int86(0x10, &regs, &regs);
regs.x.ax=0x1103;
regs.h.bl=t2;
int86(0x10, &regs, &regs);
}

```

表 4. 寄存器操作及字符表选择函数

表 5 是装入汉字字符表并选择字符表的流程图, 限于篇幅, 程序从略。

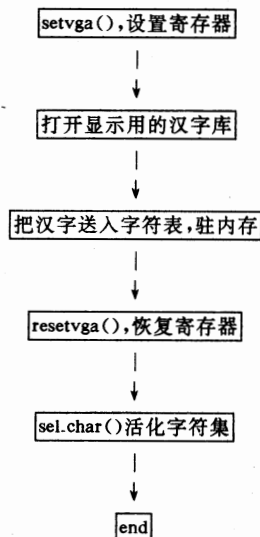


表 5

当我们在系统中装入并选择了汉字符表后, 直接对显示区内内存进行操作就可以将汉字显字在屏幕上。如果我们在系统中字符表 2 中装入了汉字字符表, 字符表 1 中装入了 ASCII 字符集, 那么表 6 所示程序将显示所有 ASCII 字符和字符表 2 中的所有汉字。

display()

```

{
    int i;
    for(i=0; i<256; i++)
        poke(VSG, i*2, i+0x1700); /* VSG: video ram segment/
    for(i=0; i<256; i++)
        poke(VSG, i*2+640, i+0x1f00);
}

```

表 6. 字符表的显示

采用上述方法, 我们用 Turbo C 在 AST386 SX/16 上实现了汉字显字的弹出式菜单。

在系统中装入汉字字符表来显示汉字的方法, 可以在装 EGA 或 VGA 的 IBM-PC 系列及其兼容机上实现, 并且可以把这种思想应用于装有其它类型显示器的计算机。使用这种方法显示汉字不占用系统内存, 汉字显字快速方便, 并且可以显示多种字体。这种方法简易方便, 可以用汇编语言和多种高级语言来实现。

巧解黑色星期五病毒

长春市东北水利电力专科用电 891 班(130012) 谭小敏

黑色星期五病毒(Jerusalem)是一种流行较广且危害较大的恶性病毒。本文介绍一个利用 DEBUG 汇编的小程序,该程序可以比较方便地对计算机内存进行黑色星期五病毒的检查 and 清除。该程序本人在长城 0520 机上调试验证,效果不错。具体实现步骤如下:

```
A>debug
-A100
XXXX 0100  MOV SI,0173
XXXX 0103  CALL 0127
XXXX 0106  MOV AX,3521
XXXX 0109  INT 21
XXXX 010B  CMP BX,025B
XXXX 010F  JZ 011A
XXXX 0111  MOV SI,01B5
XXXX 0114  CALL 0127
XXXX 0117  RET
XXXX 0118  INT 20
XXXX 011A  MOV AX,3508
XXXX 011D  INT 21
XXXX 011F  CMP BX,021E
XXXX 0123  JZ 0136
XXXX 0125  JMP 0111
XXXX 0127  CS:
XXXX 0128  LODSB
XXXX 0129  AND AL,7F
XXXX 012B  JZ 0117
XXXX 012D  MOV AH,0E
XXXX 012F  MOV BX,0007
XXXX 0132  INT 10
XXXX 0134  JMP 0127
XXXX 0136  PUSH DS
XXXX 0137  ES:
XXXX 0138  MOV AX,[0015]
XXXX 013B  MOV DS,AX
XXXX 013D  ES:
XXXX 013E  MOV DX,[0013]
XXXX 0142  MOV AX,2508
XXXX 0145  INT 21
XXXX 0147  MOV AX,3521
XXXX 014A  INT 21
XXXX 014C  ES:
XXXX 014D  MOV AX,[0019]
```

```
XXXX 0150  MOV DS,AX
XXXX 0152  ES:
XXXX 0153  MOV DX,[0017]
XXXX 0157  MOV AX,2521
XXXX 015A  INT 21
XXXX 015C  MOV AL,90
XXXX 015E  ES:
XXXX 015F  MOV [0275],AL
XXXX 0162  ES:
XXXX 0163  MOV [0276],AL
XXXX 0166  ES:
XXXX 0167  MOV [0277],AL
XXXX 016A  POP DS
XXXX 016B  MOV SI,01D3
XXXX 016E  CALL 0127
XXXX 0171  INT 20
XXXX 0173
```

```
-E173 0D 0A 54 48 45 20 50 72 6F 67 72 65 6D
-E180 20 6F 66 20 4B 69 6C 6C 20 4A 65 72 75 73 61 6C
-E190 65 6D 20 4D 61 64 65 20 42 79 20 54 61 6E 20 58
-E1A0 69 61 6F 20 4D 69 6E 21 31 30 2F 31 30 2F 31 39
-E1B0 39 31 0D 0A 00 0D 0A 4E 6F 20 4A 65 72 75 73 61
-E1C0 6C 65 6D 20 56 69 72 75 73 20 46 6F 75 6E 64 21
-E1D0 0D 0A 00 0D 0A 46 6F 75 6E 64 20 4A 65 72 75 73
-E1E0 61 6C 65 6D 20 56 69 72 75 73 21 21 4E 4F 57 20
-E1F0 69 73 20 4B 69 6C 6C 65 64 21 21 47 4F 4F 44 20
-E200 42 59 45 0D 0A 00 4D 73 44 6F 73
-N JD.COM
-R BX
BX XXXXX
:0000
-R CX
CX XXXXX
:010B
-w 100
-writing 10B Bytes
-Q
```

这样便在磁盘上生成了一个名为 JD.COM 的解毒程序,且在程序末尾加上了免疫标志,运行时只要在 DOS 状态下键入 JD 即可运行。
注意编制该程序时应应在无毒环境下进行。

通用的汉化 WordStar

四川重庆医药设计院(630042) 曹国钧

目前, IBM PC/XT, AT, 286, 386 及其各种兼容机上广泛使用汉字 WordStar, 它不仅可以编辑公文、书信, 输入高级语言的源程序, 而且也具有极强的打印控制功能。现在, 在用户手上流行两种汉化版本, 一种是机电部六所研制的汉化 WordStar 显示 10 行汉字, 适用于 IBM PC/XT 机上彩色中分辨率模式; 另一种就是中国计算机技术服务公司研制开发的汉化 WS 显示 25 行汉字, 适用于各种类型 286, 386 型微机。当今, 广大用户拥有 Super-PC 微机, 它却只能显示 21 行汉字, 如果把上述两种 WS 汉化版本运行于这种微机上, 就会出现不断闪烁跳动的现象, 给用户带来诸多不便。因此, 笔者根据长期使用汉化 WordStar 的经验, 开发了一种通用的汉化 WordStar 版本。

由于汉化 WordStar 控制行数的模块在主引导文件 WS.COM 中, 我们只需修改此文件即可。为了使汉化 WordStar 通用化, 可以显示 10 行、16 行、21 行、25 行的汉字, 我们在 WS.COM 中增加一段新的汇编语言程序, 用它来控制不同显示行数。为此, 我们对 WS.COM 作了如下修改:

对 WS.COM 修改第一条指令(注: 原第一条指令用调试程序 debug.com 的 R 命令查看, 并记下, 下面有用), 使之跳转到我们新增加的行数控制程序段, 待此程序段运行完毕后, 转移到原 WS.COM 执行段处, 即原 WS.COM 中 `××××:0100 JMP 3AB1` 代码处, 然后运行 WS.COM 全部可执行代码。这样, 既控制了不同的显示行数, 又不破坏 WS.COM 文件。

下面就是具体修改 WS.COM 的步骤:

```
C> debug WS.COM
-A
××××:0100 JMP 5600 (先用 R 命令查看 CX 中内容)
××××:0103
-A 5600
××××:5600 push ax
××××:5601 push dx
××××:5602 mov dx, 5631
××××:5605 mov ah, 09
××××:5607 int 21
××××:5609 mov ah, 01
××××:560B int 21
××××:560D cmp al, 31
××××:560F jz 5627
```

```
××××:5611 cmp al, 32
××××:5613 jz 5620
××××:5615 cmp al, 33
××××:5617 jnz 562C
××××:5619 mov byte ptr [0248], 19 (25 行)
××××:561E jmp 562C
××××:5620 mov byte ptr [0248], 14 (21 行)
××××:5625 jmp 562C
××××:5627 mov byte ptr [0248], 0F (16 行)
××××:562C pop dx (10 行)
××××:562D pop ax
××××:562E jmp 3AB1
××××:5631 DB ' 0:10 行屏幕显示', 0D, 0A
××××:5661 DB ' 1:16 行屏幕显示', 0D, 0A
××××:5682 DB ' 2:21 行屏幕显示', 0D, 0A
××××:56A3 DB ' 3:25 行屏幕显示', 0D, 0A
××××:56BC DB ' 本软件由重庆大学系统工程及
应用数学系曹国钧编改', 0D, 0A
××××:5707 DB ' 请选择屏幕显示行数<0>$'
××××:5722
-R CX
CX 5500
:5730
-W
Writing 5730 bytes
-Q
```

至此, WS.COM 修改完毕。为了使读者更好地使用 WordStar, 笔者编制了一个简单但优美的批处理文件 WS.BAT, 首先应将 WS.COM 换名为 CWS.COM, 其内容如下:

```
C> COPY CON WS.BAT
ECHO OFF
CLS
ECHO+ (注: O 与 + 之间不应留有空格, 此命令是为了使光标换行)
ECHO+
ECHO+
ECHO * * * * * 通用的汉化 WordStar 1.00 Ver 选
择行数菜单 * * * * *
CWS
ECHO ON
^ Z (按下功能键 F6)
```

于是, 完整的通用汉化 WordStar 编改完成, 读者可以根据自己的微机显示行数使用 WordStar。

反函数查找法的实现

青海第二机床厂产品开发部(810021) 王杰民

查找在计算机程序设计中占有重要地位,对有序表(以下设定按从小到大的顺序),可以采用查找效率较高的折半查找法。折半查找的基本思想是:经过一次关键字比较,将有序表分割成两部分,尔后只在表的一部分(子表)中继续进行,直到找到或找不到某记录为止,具体过程是这样的:先取表的中间位置的记录关键字与给定值比较,若相等,则查找成功;否则,若给定值比该记录的关键字值大(小),就再在表的后(前)半部分找。在表的后(前)半部分中仍取中间位置的记录进行比较,又可舍去这部分中的一半,等等。依次反复进行,直到找到或区间长度小于零表明找不到为止。

由上述可知,折半查找是对有序表基于关键字比较的方法。显然,此表是有限个记录的有序集合,但在实用上,尤其是在工程设计中,有序表的查找并非都能简单套用折半查找法,以齿轮传动设计中常见的渐开线函数为例:渐开线函数 $y = \text{tg}(x) - x$,其中自变量 x 为传动啮合角。实际问题往往是已知某一渐开线函数值,求传动啮合角的值。这一类反函数计算在工程计算中是常见的,可以用迭代法求解,也可以用数据结构中的查找技术求解。

不难看出这类求反函数问题有以下特点:

1) 只知给定值,而未给出一系列待查记录及其有序表。换句话说,预先并不知道关键字的情况。这样一来,就无法直接采用关键字比较的方法,只能分别假设自变量,根据给定公式逐个求出函数值,再将其逐一与给定函数值比较。可见,需要花费大量时间来建造待查记录的有序表,而且这个表是动态变化的;

2) 给定区间内待查记录的个数是无限的,因此函数的个数也是无限的,其值往往是浮点数。由此,判断查找是否成功就不宜进行浮点数比较,而应给出允许误差,只要函数值与给定值两者绝对值之差小于或等于允许误差,就应视为查找成功。否则,区间长度永远不小于零,查找永远不会成功。

由以上两点可见,折半查找在求解反函数问题时,对动态表结构是不适宜的。为此,可对折半查找法加以改进。

反函数查找法是面向动态表的查找方法,它与折半查找法相似。该法要点是:为简化计算,初始查找区间只设上界,不设下界,自变量的估计采取变增量,每计算完一遍,令增量缩小一半,将每遍算出的函数值与给定值比较。如两者之差超过允许值,则将自变量增加或减少一个增量,再代入公式计算,直到满足精度要求为止。问题在于,折半查找法计算的是数组元素的下标,而反函数查找法计算的是元素值本身。由于受计算机字长的限制,在折半时对超长部分必须舍入。这样一

来,不仅增加了时间开销,而且会导致自变量某些区间复盖不到,以至引起查找失败。因此我们将自变量变换为 2 的若干次幂,这样,折半运算时就不存在余数问题了,这是反函数查找法的关键所在。

为了使读者把形式描述的算法改写成程序,这里我们给出算法基本思想的描述。

```
procedure search(y,x,real);
    {y,x 分别为函数给定值与所求的自变量}
    var k,a1,b1,c1:integer;
        a12,yy,d,aq,aq1:real;
    begin
        置允许的精度要求 d;
        k:=1; {自变量的指数}
        置自变量初值上界 a1;
        b1:=a1;
        repeat
            估算自变量 a12;
            估算函数值 yy;
            aq1:=abs(y-yy);
            if aq1<=d
                then x:=a12; {查找成功}
                else c1:=b1*a(-k); {增量减半}
                k:=k+1;
                case y=yy: q:=a12; return;
                    y>yy: a1:=a1+c1;
                    y<yy: a1:=a1-c1;
                end;
        until (查找成功)
    end.
```

本算法的平均比较次数是自变量密度的函数,与表长无关。其适用范围为求反函数的场合。只要允许的精度合理,查找肯定会成功。

今举例说明。渐开线函数按下式计算

$$y = \text{tg}(x) - x$$

其中 x 为传动啮合角,一般小于 30 度,计算时以弧度值代入。

已知 $y = 0.014904383$

求 x ,要求精确到秒。若以该 x 为自变量,得出的 y 值与给定值之差小于或等于 0.0000001,则视为查找成功。

解:

30 度 = 108000 秒

初值上界可取 2 的 17 次幂 = 131072 秒

经 12 次查找得 $x = 20$ 度

答:传动啮合角为 20 度。

下面给出一个由渐开线函数求 30 度角以内传动

(转第 12 页)

在 CCDOS2.13F 支持下 dBASE III 画图技术

云浮硫铁矿企业集团公司信息中心 徐国茂 黄松德 刘少明

计算机辅助企业管理,较多的是为企业提供数据的存储、查询、统计和报表输出。为使问题更清楚、直观,要求反映问题的数据不但要以报表的形式输出,而且更需要以图形的方式来表示,即要求图文并茂。但现在普遍使用的 dBASE III 却没有提供绘图功能,笔者为满足公司所属的质监、销售、安全等用户单位的要求,通过不断摸索,找到一种在 dBASE III 状态下调用数据库数据自动在屏幕上绘制并通过硬拷贝输出企业管理中经常使用的折线图、直方图、扇形图的方法,在实际应用中效果良好。现将此方法编制成较通用的程序提供给读者参考。

1. 本文程序的运行环境

硬件:IBM-PC/XT 及其兼容机。软件:CCDOS2.13F 系统,dBASE III-PLUS,CEPSG.COM 硬拷贝程序(640*200 分辨率用 SGP.COM)。

众所周知,各种图形都是由一些不同的线组成,而线又是由最基本的点组成。所以,画点是解决问题的关键。在 CCDOS2.13F 汉字系统中,提供了一组功能特殊的显示命令(在此按 dBASE III 格式列举与问题相关的部分命令):

(1)画点: ? CHR(14)+“D 色号,点 X,Y”

(2)画横线: ? CHR(14)+“H 色号,左端点 X,Y,长度”

(3)画竖线: ? CHR(14)+“S 色号,上端点 X,Y,长度”

(4)画矩形: ? CHR(14)+“B 色号,左上角点 X,Y,宽,高”或 ? CHR(14)+“E 色号,左下角点 X,Y,宽,高”

以上这些命令的参数都要求是字符型参数。

根据此组显示命令,且以用户单位提供的数据,作为相应的 X,Y 坐标画出点,组成线,构成图形。为了使图形更美观,我们在程序中使用 ? CHR(14)+“10,160”和 ? CHR(14)+“10,161”,CHR(14)+“119,0”和 ? CHR(14)+“119,1”来取消、建立光标和取消、建立提示行,使屏幕图形整洁、稳定。

2. 画折线、直方、扇形图的程序说明

程序用 dBASE III-PLUS 编写,如果手头上只有 dBASE III,则略加修改就可以使用(去掉 INKEY() 函数,用过程代替语句 DO(文件名)WITH(参数),并去掉该语句。)

程序使用的数据库是:GYK.DBF,内容如下:

编号 (BH)	含硫量 (S)	编号 (BH)	含硫量 (S)	编号 (BH)	含硫量 (S)
01	49.65	05	49.56	09	49.69
02	49.16	06	49.24	10	49.53
03	49.48	07	49.06	11	49.30
04	49.94	08	49.67	12	49.80

(1)画折线图

①变量说明:X0,Y0 表示坐标轴原点;H:纵坐标高;L:横坐标长;M:折线的折点数;N:两折点间横坐标的距离;X1,Y1,X2,Y2 分别表示折线中某一段斜线的起始坐标和终止坐标;F:用来确定起点的位置。

②程序思想:

A:首先根据 GYK.DBF 求出第一段斜线的起点(X1,Y1)和终点(X2,Y2)

B:以 X 为自变量,步长为 0.5,根据直线方程 $Y = (Y2 - Y1) / (X2 - X1) * (X - X1) + Y1$ 作(X1,Y1), (X2,Y2)两点间的斜线。

C:画完后再调用 ZD.PRg 在斜线的终点作一较大的点。

D:求下一段斜线两端点坐标; $X1 = X2; Y1 = Y2;$
 $X2 = X1 + N; Y2 = F - (S - 49) * 180$ (Y2 的算式可根据实际需要自定)。

E:画完退出,否则转 B。

③程序清单:

```
* PROGRAM ZXT.PRg
SET TALK OFF
? CHR(14)+“1,160”
? CHR(14)+“119,0”
CLEA
USE GYK
MAX=S
MIN=S
DO WHILE NOT. EOF()
IF S>MAX
MAX=S
ENDIF
IF S>MIN
MIN=S
ENDIF
SKIP
```

```

ENDD
M=RECCOUNT()
GO TOP
N=640/M /* N 表示折线段数 */
COL0="2"
X0=10
Y0=315 /* X0,Y0 表示坐标原点 */
H=315
L=600
X1=X0
Y1=(MAX-S)*230/(MAX-MIN)+50
S1=S
C=2
E=0
@ 0,25 SAY "产品检验含硫量曲线图"
? CHR(14)+"S"+"5"+","+STR(X0)+","+STR(Y0-H)+","+STR(H)+"]"
? CHR(14)+"H"+"5"+","+STR(X0)+","+STR(Y0)+","+STR(L-5)+"]"
@ 0,1 SAY "品"
@ 1,1 SAY "位"
@ 2,0 SAY "(%) "
@ 22,(L+X0)/8 SAY "编号"
@ 0,0 SAY " "
I=1
X=X1
DO WHILE I<=M
? CHR(14)+"S5,"+STR(X)+","+STR(Y0-5)+"5]"
@ 23,X/8 SAY BH
@ 0,0 SAY " "
I=I+1
X=X+N
SKIP
ENDD
GO TOP
SKIP
DO ZD WITH C,X1,Y1
DO WHILE E=0
X=X1
IF S>=S1
? CHR(14)+"["+STR(C)+"-"+STR(X1+5)+"|"+STR(Y1+5)+"@u"+STR(S1,5,2)+"]"
ELSE
? CHR(14)+"["+STR(C)+"-"+STR(X1+5)+"|"+STR(Y1-5)+"@u"+STR(S1,5,2)+"]"
ENDI
DO WHILE .NOT.EOF() .AND. E=0
X2=X1+N
Y2=(MAX-S)*230/(MAX-MIN)+50
DO WHILE X<=X2 .AND. E=0
@ 0,0 SAY " "
Y=(Y2-Y1)/(X2-X1)*(X-X1)+Y1
? CHR(14)+"D"+STR(C)+","+STR(X)+","+STR(Y)+"]"

```

```

","+STR(Y)+"]"
X=X+0.5
E=INKEY()
ENDD
DO ZD WITH C,X,Y
X1=X2
Y1=Y2
S1=S
SKIP
IF S>=S1
? CHR(14)+"["+STR(C)+"-"+STR(X1-16)+"|"+STR(Y1+5)+"@u"+STR(S1,5,2)+"]"
ELSE
? CHR(14)+"["+STR(C)+"-"+STR(X1-16)+"|"+STR(Y1-20)+"@u"+STR(S1,5,2)+"]"
ENDI
C=C+1
IF C=8
C=2
ENDI
ENDD
GO TOP
X1=X0
Y1=(MAX-S)*230/(MAX-MIN)+50
S1=S
SKIP
ENDD
USE
? CHR(14)+"I19,1]"
? CHR(14)+"I0,160]"
RETURN

* PROGRAM ZD. PRG
PARAM C,X,Y
? CHR(14)+"D"+STR(C)+","+STR(X-1)+","+STR(Y)+"]"
? CHR(14)+"D"+STR(C)+","+STR(X+1)+","+STR(Y)+"]"
? CHR(14)+"D"+STR(C)+","+STR(X)+","+STR(Y-1)+"]"
? CHR(14)+"D"+STR(C)+","+STR(X)+","+STR(Y+1)+"]"
? CHR(14)+"D"+STR(C)+","+STR(X+1)+","+STR(Y-1)+"]"
? CHR(14)+"D"+STR(C)+","+STR(X-1)+","+STR(Y+1)+"]"
? CHR(14)+"D"+STR(C)+","+STR(X+1)+","+STR(Y+1)+"]"
RETURN

```

(2)画直方图:

①变量说明:

X0,Y0:用来确定坐标原点,Y0 还用来表示方块

的左下角纵坐标;H,L:分别表示纵横坐标长度;

SUM:表示方块图的方块数;K:表示方块宽度;G:表示方块高度(其算式可考虑字段S的值及显示器的分辨率自行确定,以免超出屏幕)。

②程序思想

A:确定第一个方块的左下角横坐标X及方块高度G。以(X1,Y0)为方块的左下角坐标,K为宽,G为高作第一个方块。

B:求下一个方块的左下角横坐标X,方块高G:X=X+K;以(X,Y0)为此方块的左下角坐标。

K为宽,G为高作方块。

C:画完退出,否则转B。

③程序清单:

```
* PROGRAM ZZT. PRG
SET TALK OFF
CLEA
? CHR(14)+"I19,0]"
? CHR(14)+"I0,160]"
USE GYK
MAX=S
MIN=S
DO WHILE .NOT. EOF()
IF MAX<S
MAX=S
ENDI
IF MIN>S
MIN=S
ENDI
SKIP
ENDO
COUNT TO SUM
D=0
C=5
X0=10
Y0=319
H= 309
L= 630
@ 0,20 SAY "产品检验含硫量直方图"
? CHR(14)+"S"+STR(C)+","+STR(X0)+","+
STR(Y0-H)+","+STR(H)+"]"
? CHR(14)+"H"+STR(C)+","+STR(X0)+","+
STR(Y0)+","+STR(L)+"]"
@ 0,2 SAY "(S%)"
@ 23,(L+X0)/8-4 SAY "编号"
DO WHILE D=0
@ 0,0 SAY " "
D=INKEY()
K=570/SUM
X=30
DO WHILE .NOT. EOF() .AND. D=0
@ 0,0 SAY " "
G=20+(S-MIN)*270/(MAX-MIN)
? CHR(14)+"E"+STR(C)+","+STR(X,10,2)+","
```

```
+STR(Y0)+","+STR(K,10,2)+","+STR(G,10,2)+
"]"
```

```
? CHR(14)+"["+STR(X+5,10,2)+"|"+STR(Y0-
G-20)+"@u"+LTRIM(STR(S,6,2))+"]"
```

```
@ 23,(X+K/2)/8 SAY BH
```

```
@ 0,0 SAY " "
```

```
L=X
```

```
DO WHILE L>=X .AND. L<X+K .AND. D=0
AND. X+K-L>=15
```

```
? CHR(14)+"S"+STR(C)+","+STR(L,10,2)+","+
STR(Y0-G,10,2)+","+STR(G,10,2)+"]"
```

```
@ 0,0 SAY " "
```

```
L=L+15
```

```
D=INKEY()
```

```
ENDD
```

```
C=C+1
```

```
IF C=8
```

```
C=2
```

```
ENDI
```

```
SKIP
```

```
X=X+K
```

```
ENDD
```

```
GO TOP
```

```
ENDD
```

```
USE
```

```
? CHR(14)+"I0,160]"
```

```
? CHR(14)+"I19,1]"
```

```
RETURN
```

(3)画扇形图:

画扇形图时首先画出圆,然后再根据 GYK. DBF 的数据按比例分块,接逆时针方面旋转作扇形。

①变量说明:

X0,Y0,R:表示圆心坐标和半径;NUM:为扇形数;SUM:为各块大小的总和;PIE:为 π 的值;A:为当前扇形与第一个扇形起始边的高角;SINA:为角A的正弦值。

②程序思想

A:以(X0,Y0)为圆心,R为半径作圆(分别以X,Y为自变量作两次)。

B:在点(X0,Y0)和点(X0+R,Y0)之间作横线,以此横线作为第一个扇形的起始边。

C:求出扇形角A。

D:根据 Maclaurin 展开式求出角A的正弦值SINA,根据SINA的值定出扇形终边与圆的交点(X1,Y1)

F:在点(X0,Y0)与(X1,Y1)之间作斜线。

G:画完退出,否则转C。

③程序清单:

```
* FKT. PRG
SET TALK OFF
? CHR(14)+"I0,160]"
PIE=3.1415926
X0=320
```



```

Y0=170
R=130
C=5
Y=Y0-R
? CHR(14)+"I19,0]"
CLEA
D=0
@ 0,30 SAY "产品含硫量扇形图"
DO WHILE D=0
DO WHILE (Y>=Y0-R) .AND. (Y<=Y0+R) .
AND. D=0
X1=X0+SQRT(R*R-(Y-Y0)*(Y-Y0))
X2=X0+SQRT(R*R-(Y-Y0)*(Y-Y0))
? CHR(14)+"D"+STR(C)+","+STR(X1)+","+
STR(Y)+"]"
? CHR(14)+"D"+STR(C)+","+STR(X2)+","+
STR(Y)+"]"
Y=Y+2.5
@ 0,0 SAY " "
D=INKEY()
ENDD
X=X0-R
DO WHILE (X>=X0-R) .AND. (X<=X0+R) .
AND. D=0
Y1=Y0+SQRT(R*R-(X-X0)*(X-X0))
Y2=Y0-SQRT(R*R-(X-X0)*(X-X0))
? CHR(14)+"D"+STR(C)+","+STR(X)+","+STR
(Y1)+"]"
? CHR(14)+"D"+STR(C)+","+STR(X)+","+STR
(Y2)+"]"
X=X+2.5
@ 0,0 SAY " "
D=INKEY()
ENDD
? CHR(14)+"H5,"+STR(X0)+","+STR(Y0)+","
+STR(R)+"]"
USE GYK
COUNT TO NUM
SUM S TO SUMM
GO TOP
E=0
DO WHILE .NOT. EOF() .AND. D=0
A=S/SUMM*2*PIE+E
B=E+(S/SUMM*2*PIE)/2
M=1
SINA1=10
SINB=0
SINA=0
T=.T.
DO WHILE T.AND. D=0
D=INKEY()
I=1
SIN=1
SINBB=1
ZC=1

```

```

DO WHILE I<2*M .AND. D=0
D=INKEY()
ZC=I*ZC
I=I+1
ENDD
I=1
DO WHILE I<2*M .AND. D=0
D=INKEY()
SIN=SIN*A
SINBB=SINBB*B
I=I+1
ENDD
IF MOD(M,2)=0
NN=-1
ELSE
NN=1
ENDI
SINA=SINA+NN*SIN/ZC
SINB=SINB+NN*SINBB/ZC
M=M+1
IF ABS(ABS(SINA)-ABS(SINA1))<0.00001
T=.F.
ENDI
SINA1=SINA
ENDD
Y1=Y0-R*SINA
YB=Y0-R*SINB
IF (A>PIE/2).AND.(A<3*PIE/2).AND. D=0
X1=X0-SQRT(R*R-R*R*SINA*SINA)
ELSE
X1=X0+SQRT(R*R-R*R*SINA*SINA)
ENDI
IF (B>PIE/2) .AND. (A<3*PIE/2)
XB=X0-SQRT(R*R-R*R*SINB*SINB)
ELSE
XB=X0+SQRT(R*R-R*R*SINB*SINB)
ENDI
IF X1>X0
X=X0
MAX=X1
ELSE
X=X1
MAX=X0
ENDI
IF ABS(X1-X0)>1
DO WHILE X<=MAX .AND. D=0
@ 0,0 SAY " "
Y=(Y1-Y0)/(X1-X0)*(X-X0)+Y0
? CHR(14)+"D"+STR(C)+","+STR(X)+","+STR
(Y)+"]"
X=X+0.9
D=INKEY()
ENDD
ELSE
IF A>PIE

```

```

? CHR(14)+"S"+STR(C)+","+STR(X0)+","+STR(Y0)+","+STR(R)+""]
ELSE
? CHR(14)+"S"+STR(C)+","+STR(X0)+","+STR(Y0-R)+","+STR(R)+""]
ENDI
ENDI
IF XB>=X0 .AND. YB<=Y0
? CHR(14)+ "["+STR(XB+15)+"|"+STR(YB-15)
+["@u"+LTRIM(STR(S,8,2))+""]
ELSE
IF XB<=X0 .AND. YB<=Y0
? CHR(14)+ "["+STR(XB-10*LEN(LTRIM(STR(S,
8,2))))+"|"+STR(YB-15)+["@u"+LTRIM(STR(S,8,
2))+""]
ELSE
IF XB>=X0 .AND. YB>=Y0
? CHR(14)+ "["+STR(XB+5)+"|"+STR(YB+5)+
+["@u"+LTRIM(STR(S,8,2))+""]
ELSE
IF XB<=X0 .AND. YB>=Y0

```

```

? CHR(14)+ "["+STR(XB-10*LEN(LTRIM(STR(S,
8,2))))+"|"+STR(YB)+["@u"+LTRIM(STR(S,8,2))+"
"]
ENDI
ENDI
ENDI
SKIP
E=A
C=C+1
IF C=8
C=2
ENDI
ENDD
ENDD
USE
? CHR(14)+"I19,1"
? CHR(14)+"I0,160"]
RETU

```

〔编者按：本文选自 1991.11 广东省计算机学会数据库技术交流会论文集〕

(接第 2 页)

盘的力量。总的来说生理特点校验比行为特点校验更为可靠,但所需投资也相应要高得多。图 1 是一种校验视网膜血管布局的示意图,光在视网膜上进行一次 450 度扫描,对其上的 320 个数据点进行验证——即使是极为相似的双胞胎也不会完全相同。现在美国市场上出售的生理校验设备有 8 种,另有 30 种正在研制中。但因其售价太高(600~7000 美元),目前计算机用户还无法普遍使用。表 1 是美国生理和行为校验设备销售情况,其中售出的设备仅有 16% 用于计算机安全,据报道近几年来这一百分比正在上升。

表 1

检验设备	售出数量		平均售价		最低价
	1987 年	1988 年	1987 年	1988 年	1989 年
指纹检验机	260	475	7000	4200	1800
视网膜扫描仪	125	175	7504	7000	5000
手形检验机	60	130	4200	3300	3000
声音校验	600	625	1200	1200	1200
动态签字辨别	125	195	800	800	640
动态击键辨别	80	0	700	700	600
总和	1250	1600			

注:售价为美元

(接第 7 页)

啮合角的实用的 BASIC 程序,读者据此很容易改写成求解具体问题的程序。

反函数查找程序

```

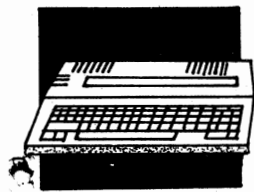
10 INPUT Y#
20 GOSUB 50
30 PRINT X# * 57.295779506#
40 END
50 'search(y,x)
60 D=.0000001
70 K=1
80 AL=131072#
90 BL=AL
100 AL2# =AL * 3.141592654# /180/3600

```

```

110 YY# =TAN(AL2#) -AL2#
120 AQ# =Y# -YY#
130 AQ1=ABS(AQ#) -D
140 IF AQ1<=0 THEN 230
150 CL=BL * 2 ^ (-K)
160 K=K+1
170 IF AQ# >=0 THEN 200
180 AL=AL-CL
190 GOTO 100
200 IF AQ# =0 THEN 230
210 AL=AL+CL
220 GOTO 100
230 X# =AL2#
240 RETURN

```



学习机之友

英语单词快速记忆系统

天津第七印刷厂设备科(300061) 张振堂

计算机教学的特点有利于因材施教。从教育心理学的角度讲,每个人有不同的情绪、智力,一位教师面对数十位学生,要清醒地掌握他们的智力情况,不是一件容易事。这里,计算机教学的优点就显露出来了。例如:计算机可出题供学习者练习,练习题可根据学生的回答内容动态生成,还能根据学生回答问题的“表现”,“诊断”出学生困难的根源,从而达到对学生逐个“精雕细刻”的目的。

“英语单词快速记忆系统”在这方面做了初步尝试,获得了较好的效果,现将它奉献给读者。

一、设计思想

本系统主要依据记忆规律,合理控制单词重复的时间间隔。由记忆特点得知:人们记忆时开始一段时间遗忘较快,随着时间的推移,遗忘逐渐放慢,最后记忆效果得到巩固。因此,欲使生词在遗忘前得到巩固,应确保其出现的频数,而熟词过多的出现不但浪费练习时间,也分散学生的精力,应受到控制。然而,这不应意味着对熟词的忽视。此外,某些人对某些特定的单词接受较快或较慢的现象也是有的。对此应加强系统对这些信息的反馈。

根据以上要求,本系统将附设的单词库分为生词库和熟词库,生词库中存放用户输入的生词。熟词存放已经学会的单词,根据学生对其熟悉程度又被分为九级。练习时,生词库中单词出现的概率最高,熟词库中单词出现的概率随单词级别的提高而递减,任何单词回答正确都可逐级上升到较高一级的词库,若答错了则不管原来级别都降回生词库。这种结构可保证充分练习生词而兼顾复习熟词,用最短的练习时间获取最佳的学习效果。

为了确保学习的连续性及有效性,系统词库内全部信息都可以脱机保存,以便任何时间开机都可接续上次的练习,并达到同样的学习效果。

二、主要结构及功能

本系统有多种功能,在一个汉字主菜单控制下进行选择操作,使用极为简单。根据菜单的提示,按一键即可转入相应的功能块。现分别叙述如下:

(1)查看单词库:此项功能可浏览指定级别词库内的单词及有关信息。学生可以由此找出自己的薄弱环节。教师也可以据此总结教学的规律。

(2)输入新单词:根据屏幕提示依次键入单词、词性及汉译,键入过程中屏幕将自动变换状态(字母、拼写)以方便操作。拼写单词时可大写也可小写;使用时应按照输入方式回答。拼写完毕,程序将对系统词库内

单词进行检索,只有词库内没有重复单词时,该单词才会被接纳进入生词库。

(3)删除旧单词:进入此项功能后,屏幕显示一个子菜单,可完成以下子功能:

i 删除指定级别单词:随着学生学习水平的提高,某些级别的单词已被记牢,我们可以将其删除,腾清词库空间,以便容纳新单词。

ii 删除指定单词:输入生词时,如果拼写出现错误,可通过此项功能进行删除,以便重新输入。

iii 删除全部单词:这里,词库内单词全部被删除,因此应慎用此功能。

(4)汉译英练习:由用户指定练习次数,然后开始练习,练习过程中将充分体现前文所叙各项特点。最后,练习结果将显示于屏幕。

(5)统计:词库内各级别单词的数量将被分别统计出来以供参考。我们可以据此了解学生对全部单词的掌握情况。

(6)退出:词库内全部单词及学习者对单词的掌握情况被编制成 DATA 语句,然后系统对使用磁盘机还是磁带机进行鉴别,若是磁盘机,这些数据将会自动存入磁盘,若是磁带,屏幕会提示用户操作录音机以便将数据录入磁带。

三、使用说明

这里对使用中可能出现的问题作几点说明:

如使用磁盘作外存,应使磁盘内含有以“ENGLISH”为名的本程序,否则由于寻不到该文件将会出现错误。如果欲改为其它文件名,用户可对 5 号语句的 U\$ 值进行修改即可。

由于本系统的特点在于针对每个学习者进行个别指导,所以不可数人共用一个程序,否则系统的优越性难以发挥。如有 A、B 二人学习,可以将程序分别命名为 ENGLISH-A 及 ENGLISH-B,用户同样应对程序中 U\$ 值进行修改。此后练习中,A、B 二人只可调用磁盘同名程序。

只有经常使用本系统进行练习,系统内才会积累有关你的大量资料,才能了解你,也才能有的放矢地帮助你。因此,只有经常和它交谈,不冷落它,它才会成为你的好助手。

本程序在 CEC-I 机上通过。

```
5 LOMEM,24576;E$=CHR$(18);D$=CHR$(4);F=0;
  L=0;V$="请等待...";U$="ENGLISH"
10 REM DISK SYSTEM?
```



```

15 DK=PEEK(994)+PEEK(1001)-192
25 IF DK=0 THEN PRINT D$;"PR#3":GOTO 35
30 PR#3
35 PRINT:PRINT E$;HOME
40 VTAB 5:HTAB 3:PRINT "*** 英语单词快速记忆系统
   ***"
45 POKE 1013,76:POKE 1014,23:PRKE 1015,3:POKE 6,
   231:POKE 7,3
50 FOR I=0 TO 114:READ P:POKE 768+I,P:NEXT
55 READ N
60 DIM A%(9),A$(3,N)
65 IF N=0 THEN 80
70 FOR I=1 TO N:FOR J=0 TO 3:READ A$(J,I):NEXT
   J,I
80 HOME:P=FRE(0):HTAB 7:PRINT("(本词库内共含单
   词";N;"个)")
85 PRINT:HTAB 5:PRINT"请选择:":PRINT
90 HTAB 6:PRINT"[1]查看单词库"
95 HTAB 6:PRINT"[2]输入新单词"
100 HTAB 6:PRINT"[3]删除旧单词"
105 HTAB 6:PRINT"[4]汉译英练习"
110 HTAB 6:PRINT"[5]统计"
115 HTAB 6:PRINT"[6]退出";
120 GOSUB 625:IF P<1 OR P>6 THEN 120
125 IF N=0 AND(P=1 OR P=3 OR P=5)OR N<2 AND P
   =4 THEN 120
130 HOME:ON P GOSUB 145,200,275,395,520,575
135 GOTO 80
140 REM 查看
145 VTAB 4:HTAB 5:PRINT"请键入词库级别:(0~9)"
150 GOSUB 625:IF P<0 OR P>9 THEN 150
155 HOME:T=0:A$=STR$(P):Q$=CHR$(127)+
   CHR$(31)+CHR$(46+P)+"级词库"
160 FOR I=0 TO 3:VTAB 11:HTAB 8+I*2:PRINT MID
   $(Q$,3*I+1,3):NEXT
165 FOR I=1 TO N
170 IF A$<>A$(3,I) THEN 185
175 T=T+1:VTAB T:HTAB 1:PRINT A$(0,I):TAB(16);
   A$(1,I);" ";A$(2,I)
180 IF T=10 THEN T=0:GOSUB 625:HOME
185 NEXT
190 GOSUB 625:HGR2:RETURN
195 REM 输入
200 PRINT E$:VTAB 2
205 INPUT"请键入英语单词: ";Q$:VTAB 2:PRINT" "
210 IF N=0 THEN 230
215 FOR I=1 TO N
220 IF A$(0,I)=Q$ THEN PRINT:PRINT"词库内已有此
   单词.":GOTO 250
225 NEXT
230 N=N+1:GOSUB 645:A$(0,N)=Q$:A$(3,N)="0"

```

```

235 PRINT:INPUT"请键入单词词性:":A$(1,N):VTAB 4:
   PRINT" "
240 POKE 942,140:PRINT E$,E$
245 INPUT"请键入汉语译文:":A$(2,N):VTAB 6:PRINT
   " "
250 PRINT:PRINT"是否还输入新单词?"(Y/N)
255 POKE 942,129:PRINT E$
260 GOSUB 625:IF P=41 OR P=73 THEN HOME:GOTO 200
265 RETURN
270 REM 删除
275 HTAB 8:PRINT"请选择:":PRINT
280 PRINT:HTAB 6:PRINT"[1]删除指定分值单词"
285 PRINT:HTAB 6:PRINT"[2]删除指定单词"
290 PRINT:HTAB 6:PRINT"[3]删除全部单词"
295 GOSUB 625:IF P<1 OR P>3 THEN 295
300 HOME:ON P GOSUB 310,335,385
305 RETURN
310 PRINT:HTAB 5:PRINT"大于指定分值的单词将被删
   除。"
315 PRINT:HTAB 5:INPUT"指定分值=?":Q
320 T=0:FOR I=1 TO N
325 IF VAL(A$(3,I))<Q THEN T=T+1:FOR J=0 TO 3:
   A$(J,T)=A$(J,I):NEXT
330 NEXT:N=T:GOSUB 645:RETURN
335 PRINT:HTAB 5:PRINT"请键入欲删除单词:"
340 PRINT:HTAB 8:INPUT"——":Q$
345 FOR I=1 TO N
350 IF A$(0,I)=Q$ THEN 360
355 NEXT:GOTO 380
360 N=N-1:IF I=N+1 THEN 380
365 FOR K=I TO N:FOR J=0 TO 3
370 A$(J,K)=A$(J,K+1):NEXT J,K
380 GOSUB 645:RETURN
385 N=0:GOSUB 645:RETURN
390 REM 练习
395 PRINT"      共  个"
400 PRINT"      "
405 PRINT"      第  个"
410 VTAB 5:HTAB 9:INPUT"单词个数=?":Q
415 VTAB 5:PRINT CHR$(26)
420 VTAB 1:HTAB 30:PRINT Q:R=0
425 FOR I=1 TO Q
430 K=0
435 K=K+1
440 H=INT(RND(1)*N)+1
445 A=VAL(A$(3,H)):IF A>K/2 THEN 435
450 IF H=H1 THEN 440
455 H1=H:VTAB 3:HTAB 30:PRINT I
460 VTAB 2:HTAB (3+(21-LEN(A$(1,H))-2*LEN(A
   $(2,H))/3)/2):PRINT A$(1,H);" ";A$(2,H)
465 VTAB 5:HTAB 2:INPUT"请译成英语——":Q$

```

```

470 IF Q$ = A$(0,H) THEN R = R + 1; A$(3,H) = STR
    $(A + (A < 9)); MUSIC 85, 45; MUSIC 50, 50; MUSIC
    80, 50; GOTO 485
475 A$(3,H) = "0"; PRINT; HTAB 2; PRINT "正确答案是
    ——"; A$(0,H)
480 VTAB 10; HTAB 8; PRINT "按任何键继续进行"; GO-
    SUB 625
485 VTAB 2; HTAB 3; PRINT SPC(21)
490 VTAB 4; PRINT CHR$(11)
495 NEXT
500 HOME; VTAB 4; HTAB 6; PRINT "正确:"; R
505 PRINT; HTAB 6; PRINT "错误:"; Q-R
510 GOSUB 625; RETURN
515 REM 统计
520 VTAB 6; HTAB 12; PRINT V$
525 FOR I=0 TO 9; A%(I)=0; NEXT
530 FOR I=1 TO N
535 A=VAL(A$(3,I)); A%(A)=A%(A)+1
540 NEXT; HOME
550 FOR I=0 TO 9
555 VTAB I+1; HTAB 1
560 PRINT I; "级单词库....."; A%(I);
565 NEXT; GOSUB 625; RETURN
570 REM 退出
575 VTAB 6; HTAB 12; PRINT V$
580 CALL 786; N$ = STR$(N); &N$; IF N=0 THEN 590
585 FOR I=1 TO N; A$ = A$(0,I) + " " + A$(1,I) + " "
    + A$(2,I) + " " + A$(3,I); &A$; NEXT
590 P=PEEK(25)+256*PEEK(26)+2; POKE 176, INT(P/
    256); POKE 175, P-256*INT(P/256)
595 IF N=0 THEN POKE P-1, 0; POKE P-2, 0

```

```

600 IF DK=0 THEN PRINT D$; "DELETE"; U$; PRINT D
    $; "SAVE"; U$; TEXT; HOME; END
605 HOME; PRINT; PRINT "请将磁带装入带仓, 使录音机处
    于录音状态, 再按回车键。"
610 GOSUB 625; IF P<>-35 THEN 610
615 SAVE; TEXT; HOME; END
620 REM SUB-1
625 P=RND(1); P=PEEK(-16384)
630 IF P<128 THEN 625
635 POKE -16368, 0; P = P-176; RETURN
640 REM SUB-2
645 F=27+PEEK(107)+256*PEEK(108); L=21+12*N
650 POKE F+2, L-256*INT(L/256); POKE F+3, INT(L/
    256)
655 POKE F+5, INT((N+1)/256); POKE F+6, N+1-256
    *INT((N+1)/256)
660 POKE 109, F+L-256*INT((F+L)/256); POKE 110,
    INT((F+L)/256)
665 RETURN
670 DATA 165, 6, 133, 80, 165, 7, 133, 81, 32, 26, 214, 160, 1,
    177, 155, 133, 26, 136, 177, 155, 133, 25, 96, 32, 227, 223,
    230, 6, 208, 2, 230, 7, 160, 4, 169, 131, 145, 25, 136, 165, 7,
    145, 25, 136, 165, 6, 145, 25, 177, 131, 133, 28, 136, 177, 131,
    133, 27, 136,
675 DATA 177, 131, 133, 31, 24, 169, 5, 101, 25, 133, 29, 152,
    170, 101, 26, 133, 30, 177, 27, 145, 29, 200, 196, 31, 208, 247,
    138, 145, 29, 200, 145, 29, 200, 145, 29, 136, 24, 152, 101, 29,
    160, 0, 145, 25, 72, 200, 138, 101, 30, 145, 25, 133, 26, 104,
    133, 25, 96
999 REM 词库
1000 DATA 0

```

大量供应显示器及大功率开关电源

我部现货供应进口 19" 彩色、单色高分辨 (1280×1024) 图形显示器; 12" 终端, 并供应大功率开关电源, 规格有: 5V/18A, 5V/20A, +5V/50A, +5V/100A, 5V/120A, 5V/150A×2, 12V/15A。并有各种进口电脑装配电线、电缆 (单层屏蔽、多层屏蔽、单芯、多芯、镀银等) 品种繁多, 适合各类科研开发、生产、装配、工业自动化等单位使用。

以上产品均全部进品并通过美国 UL 安

全标准认证, 产品批发价相当优惠、欢迎来函、来电或来人洽谈。

广东省四会县南方电子厂经营部

地址: 广东省四会县城高观东路 71 号

邮编: 526200

电话: (07663) 322686

电挂: 1311

用 CHR \$ 函数压缩存盘数据

厦门教育学院(361003) 梁才柱

随着微机在我国的普及,“教育测量”的推广和实施已提到日程上来了。运用目前在中小学中持有量很大的中华学习机来实现“教育测量”所要求的统计与分析工作,实践证明是可行的,但需要解决好大量的统计数据与内存和磁盘容量不足这个突出问题。在编写“试题分析”软件的过程中,为了提高磁盘存储空间的利用率,本人使用过 CHR \$ 函数与 ASC 码函数来实现压缩记录成绩数据的存储空间的办法。结果表明,这种方法不但可以有效地节约磁盘的存储空间,而且使用简便,适当加强,还可以用于其他数据的压缩。为提供交流,兹将本方法介绍如下。

数据一般是使用文本文件存盘的。存盘时是以一个编码代表一个数字的方式写入。例如:“108”这个数目,在盘上是以“B1 B0 B8”三个编码存储,并且以回车(8D)作为数据间的分隔。这样,一个“108”就占用四个字节的空间。若能另选用一套有规律的代码,它只需用一个代码就能代表一个数据(例如 108)来存盘,并且取消分隔用的“8D”,那么一个数据将只占用一个字节,就可以节省 3/4 的空间。当然,就其整体来说,并不是所有的数据都是三位数,加上其他一些原因,实际节约的空间大约是 1/2 左右。但由于数据量很大,这仍将是一个很可观的数目。

选用什么代码?有没有现成的代码可供选用?回答是肯定的,那就是 CHR \$ 函数与 ASC 码函数。我们知道,CHR \$ 函数可以产生一个以给定的数值为其 ASCII 编码的字符,虽然其中有一部分是不可显示的字符,但代表该字符的编码是存在的,可以写入磁盘。而 ASC 码函数的作用恰好相反;它是产生一个用十进制数表示的字符的 ASCII 码。因此,我们就可以将数据(例如 108)转化为相应的 CHR \$ 给定值,以其编码存入磁盘中。当需要将数据读出时,便利用 ASC 码函数作为解码用,还原出相应的以十进制表示的数目。由于上述可供写入磁盘的编码共有 128 个,而目前的考试评分,最多只有 120 分。即使由于某些原因需要扣除编码中的几个,仍然是够用的。

为了具体说明上述方法的使用及其节约磁盘空间的效果,请运行以下两段小程序。其中:程序一是用普通方法存盘,程序二是用本文介绍的压缩法存盘,两者都是向磁盘写入 0—127 的 128 个数字。程序中的 60 句是两种存盘法差别之所在。

程序 1

```
10 INPUT "FILE NAME?";F$
20 D$ = CHR $(4)
30 PRINT D$;"OPEN";F$
```

```
40 PRINT D$;"WRITE";F$
50 FOR I=0 TO 127
60 PRINT I
70 NEXT
80 PRINT D$;"CLOSE";F$
100 END
```

程序 2

```
10 INPUT "FILE NAME?";F$
20 D$ = CHR $(4)
30 PRINT D$;"OPEN";F$
40 PRINT D$;"WRITE";F$
50 FOR I=0 TO 127
60 PRINT CHR $(I+5);
70 NEXT;PRINT
80 PRINT D$;"CLOSE";F$
100 END
```

现在如果用 COPY II PLUS 的 SECTOR EDITOR 功能来观察一下已写入磁盘的内容。可以看出:程序一使用了盘上的 1.5 个扇区,而程序二只用了约半个扇区,短短的 128 个数据,占用空间的差别是如此之大,压缩法的优越性是显而易见的。

最后,有几点需要说明一下:

1. 实际编程时,先用数组寄存逐个输入的成绩,然后再通过上述 60 句转化为 CHR \$ 编码存入磁盘。

2. 为了取消数据间的分隔符(8D),CHR \$(I+5) 后的“;”一定要加上,否则将存入与数据量相等的“8D”,白白多占用了一倍的空间。

3. 因此,在读出时最好也用数组接受逐个读出的数据。而且要用 GET 语句而不是用 INPUT 语句,方能把数据逐个读出。

4. 由于磁盘操作时需要用到 CHR \$(4),为防止会影响正确的存取操作,应避开它。所以要在成绩上加 5,即“CHR \$(I+5)”句,使“0”这个数从“CHR \$(5)”开始而不是“CHR \$(0)”。为此,在读出时也应该减去 5,才能得到正确的读数。例如:

```
10 FOR I=0 TO 127
20 GET A$
30 E(I)=ASC(A$)-5
40 NEXT
```

5. 以上存取的数据,均为正整数。记录成绩一般不需考虑负数。若要求比较精确的成绩记录,例如有时需要出现半分的场合,可利用“121”(即“CHR \$(126)”,编码为 FE)的编码作为半分的代码而记录入盘,取出时相应还原即可。

PB-700 微机解密技巧

山东兖州环保局(272000) 孙 力

PB-700 微机是一种物美价廉的具有广泛用途的袖珍式计算机。该机所规定的 BASIC 语言中,有专门起保护程序作用的 PASS 指令。当程序加上了任意 8 个字节之内的 PASS 通行令后,程序就进入被保护状态,只能被使用而无被破坏之虞。

本文现介绍一种解开 PASS 通行令的方法,供大家在研究、修改和应用该计算机程序时参考。

首先将所需要解密的程序由 FA-10 磁带机调入主机 P0 区,再换一 P 区调入 CASIO 公司原装磁带 GRAPH SOFTWARE 中的绘图程序(A 面,从计数器 0 开始,屏幕显示为 GRAPH PF B,大约到 40 结束,调入时若打开监听则应听到 131、311 的乐音)。

调毕用 NEWALL 命令将程序全部清除,然后再调入事先任意编写的未加 PASS 通行令的程序于主机。注意:一定要将此编写好的程序在输入磁带时将其中途破坏,即在 SAVE 此程序时中途按停止键或 BRK 键,亦可将此程序输入磁带后再用别的程序在其中输入,使其破坏。当调入该未经 PASS 的程序到显示 RW error 时,原已加密的程序就解密恢复在主机里。若用 SAVEALL 命令输入的加密程序可反复在各区使用此方法解密。

双人百米赛跑

南京梅园中学(210001)汪 波

运行程序后,先输入两位游戏者的姓名,接着,屏幕中央将出现一把发令枪,枪响后,两位游戏者就可分别通过按(Z,X)和(M,,)两键来控制两运动员的“脚”了。

此游戏最大的特点就是可两人同时玩,且互不干扰,不会出现一方按住键另一方走不了的情况。真正的体现了“竞争性”。

游戏结束时,计算机将打印出冠军,并为其奏乐,祝大家个个成为赛跑好手。

```
10 INPUT "NO. 1,NAME";A$:INPUT "NO. 2,NAME";B$:
  MODE(1);COLOR 2
20 FOR X=0 TO 127;SET(X,20);SET(X,31);SET(X,42);
  NEXT
25 FOR Y=21 TO 41;SET(127,Y);NEXT
30 A=1;B=2;Y1=25;C=1;D=2;Y2=36;COLOR 4
32 SET(60,6);SET(67,6);FOR I=60 TO 66;SET(I,7);
  NEXT;SET(64,8)
33 FOR I=8 TO 10;SET(65,I);NEXT;FOR I=8 TO 11;SET
```

```
(66,I);SET(67,I);NEXT
```

```
34 SOUND 10,2;FOR I=1 TO 400;NEXT;SOUND 15,2;FOR
  I=1 TO 300;NEXT
```

```
35 POKE 30862,74;POKE30863,52;H=USR(0);SOUND3,
  1;H=USR(0)
```

```
40 SET(A,Y1);SET(B,Y1+1);SET(C,Y2);SET(D,Y2+1)
```

```
41 IF A=127 OR C=127,100
```

```
45 P=PEEK(26875);K=PEEK(26863)
```

```
50 IF A>B AND K=223,RESET(B,Y1+1);B=B+2
```

```
60 IF A<B AND K=247,RESET(A,Y1);A=A+2
```

```
70 IF C>D AND P=239,RESET(D,Y2+1);D=D+2
```

```
80 IF C<D AND P=253,RESET(C,Y2);C=C+2
```

```
90 GOTO 40
```

```
100 SOUND 25,5;22,3;27,3;29,4;26,5;29,6
```

```
110 CLS;IF A=127,PRINT@230,A$;"WIN!";GOTO 130
```

```
120 IF C=127,PRINT @230,B$;"WIN!";PRINT
```

```
130 INPUT "PLAY(Y/N)";K$;IF K$="Y",MODE(1);
```

```
  COLOR2;GOTO 20
```

```
~~~~~
```

寻找莱蒙托夫游戏规律

南京金陵职大计算中心(210001) 钱雁群

俄国著名诗人莱蒙托夫可以让人随便想出一个自然数,然后乘以 7,把所得的积的各位数字加起来,接着乘以 9,再把所得积的各位数字加起来,最后乘以 5,除以 3,加上 5,所得结果都是 20!

我们把他的算法剥去外壳,就可发现其中奥秘。他是把一个自然数横加、乘 9 以后再横加,结果必然是 9 (横加就是把一个自然数的个位、十位、百位……分别当作个位数横向求和)。这个游戏是利用九九乘法表中的一条规律:凡是表上乘数为 9 的积,经过一次横加之,都会得 9。例如 $3 \times 9 = 27$ 。27 横加是 9。

莱蒙托夫方法是:先外套再横加,接着乘以 9,后来再横加一次。最后还有一个外套,当然是变化无穷,次次都准。

我有一个怪想法,反正要经过两次横加,不如外壳之后,先乘以 9,然后再来两次横加。结果会都等于 9 吗?手工验证几个数还行,成千上万个数的验证还得靠计算机。于是编了一个程序。

```
10 INPUT "起始数";W;INPUT "终止数";S
30 FOR A=W TO S
  40 B=A*9;PRINT A;"*9=";B,
  60 B$=STR$(B);C=LEN(B$);D=0
  90 FOR I=1 TO C
    100 D$=MID$(B$,I,1);D=D+VAL(D$)
  120 NEXT I
  130 PRINT D;" ";
  140 IF D>9 THEN B$=STR$(D);GOTO 70
  150 PRINT;NEXT A
RUN
```

起始数 20

终止数 29

20 * 9 = 180	9
21 * 9 = 189	18 9
22 * 9 = 198	18 9
23 * 9 = 207	9
24 * 9 = 216	9
25 * 9 = 225	9
26 * 9 = 234	9
27 * 9 = 243	9
28 * 9 = 252	9
29 * 9 = 261	9

RUN

起始数 99999998

终止数 100000002

99999998 * 9 = 899999982	72 9
99999999 * 9 = 899999991	72 9
100000000 * 9 = 900000000	9
100000001 * 9 = 900000009	18 9
100000002 * 9 = 900000018	18 9

本程序在中华学习机上通过,只要输入要验证的范围即起始数和终止数,计算机便可将这些数乘以 9 再来两次横加,结果应该是 9。

得到了验证,你就可以大胆出题了。我们可以让小朋友任想一个自然数,乘以 9 再横加两次,得到 9,为迷惑他人,在 9 的基础上,有意进行一些四则运算,然后报出结果,这个结果可以恰好是今天的日期或某个小朋友的年龄,朋友,你说这样好玩吗?

有一点请注意:小朋友想出的数一般都不太大,在 99999999 以内的数两次横加即可。如果和大人玩,可能要三次横加哟!

也谈“一题多解”

江苏海安县中学(226600) 郑明达

我想你肯定见过这样一道题目:从键盘接受三个整数给变量 A、B、C,然后求出它们中的最大者并输出。针对这一道题目,不少书上给出了这样的答案:

```
10 INPUT "A,B,C=";A,B,C
20 IF B>A THEN A=B
30 IF C>A THEN A=C
40 ? "MAX=";A
50 END
```

如果规定只能用一行程序完成,那该怎么办呢?

请分析一下上面这个程序,你可以发现它的基本思路是:首先,把 A 看成是三数中最大者,然后比较 A 与 B 的大小,如果此时 A 比 B 小,那么把 B 的值给 A,再用此时 A 的值与 C 比较,如果 A 的值比 C 小,那么把 C 的值给 A(即在比较过程中始终保持 A 为较大

者)。最后,输出 A 的值。

概括起来讲主要有两步:第一步找 A 与 B 中的较大者给 A。第二步找此时 A 与 C 的较大者给 A。其实质性步骤是:找两数中的较大者。针对此分析,我想用一行程序解决此问题至少有三种方法:

一、利用符号函数

大家知道式子 $(A+B+\text{SGN}(A-B)*(A-B))/2$ 可以取出 A 与 B 中的较大者。利用这个式子要写出一行程序也就十分容易啦!

```
10 INPUT "A,B,C=";A,B,C:A=(A+B+\text{SGN}(A-B)
* (A-B))/2:A=(A+C+\text{SGN}(A-C) * (A-C))/
2;? "MAX=";A;END
```

二、利用逻辑运算

大家知道,如果 $A=3, B=2$,那么 $A>B$ 的结果就是 1(逻辑真),反之为 0。则式子 $(A>=B)*A+(B>A)*B$ 不就可以取出 A 与 B 中的较大者吗? 写出的程序如下:

```
10 INPUT "A,B,C=";A,B,C:A=(A>=B)*A+
(B>A)*B:A=(A>=C)*A+(C>A)*C;?
"MAX=";A;END
```

三、利用循环语句

大家都知道:当循环 FOR I=1 TO 10;NEXT 执行后 I 的取值为 11,即 10+1;循环 FOR I=10 TO 1;NEXT 执行后 I 的取值也是 11,即 10+1。

从上述所述,我想你很快会看懂下面的程序。

```
10 INPUT "A,B,C=";A,B,C:FOR I=A TO B:
NEXT:A=I-1;FOR I=A TO C;NEXT:A=I-
1;? "MAX=";A;END
```

最后,请大家思考:如果要输出的不只是数值,还要输出变量名,该怎么修改程序?

(接第 21 页)

* BEFE;28 (以上设计 40 个磁道的磁盘)

* B5FA;15

* B4BC;15

* B293;15

* B3BC;15

* AEDC;15

* AE9F;15

* AC01;15 (十六进目录道,用户可自定)

* AEC5;60 (4 * 目录道数)

* AEC9;64 (4 * 目录道加 4)

(以上是设计目录道为 \$15,具有 40 个磁道功能的磁盘)

* ADD;BD(封锁文件名)

(3)CTRL-C (退出监控系统)

POKE 40503,0 (封锁 RESET 功能)

(4)键入 HELLO 内容

(5)INIT HELLO (格式化一张目录磁道为 \$15,40 个磁道,封锁文件名,封锁 RESET 功能的加密盘)

如在 BASIC 程序的前面加入:

```
0 ONERR GOTO 63999
```

```
63999 GOTO 63999
```

封锁 CTRL-C,(Quit)功能后加密效果更好。


```

--+
106 PRINT " |  |各|",SPC(8-LEN(STR$(MF*.8-1))),
MF*.8-1,"-";MF*.7;SPC(9-LEN(STR$(MF*.
7))),"|";SPC(4-LEN(STR$(F3))),F3;"人|";SPC
(5-LEN(STR$(FC))),FC;"%|"
107 PRINT " |数| +-----+-----+
--+
108 PRINT " | |个|",SPC(8-LEN(STR$(MF*.7-1))),
MF*.7-1,"-";MF*.6;SPC(9-LEN(STR$(MF*.
6))),"|";SPC(4-LEN(STR$(F4))),F4;"人|";SPC(5
-LEN(STR$(FD))),FD;"%|"
109 PRINT " |分| +-----+-----+
--+
110 PRINT " | |分|",SPC(8-LEN(STR$(MF*.6-1))),
MF*.6-1,"-";MF*.5;SPC(9-LEN(STR$(MF*.
5))),"|";SPC(4-LEN(STR$(F5))),F5;"人|";SPC
(5-LEN(STR$(FE))),FE;"%|"
111 PRINT " |布| +-----+-----+
--+
112 PRINT " | |数|",SPC(8-LEN(STR$(MF*.5-1))),
MF*.5-1,"-";MF*.4;SPC(9-LEN(STR$(MF*.
4))),"|";SPC(4-LEN(STR$(F6))),F6;"人|";
SPC(5-LEN(STR$(FF))),FF;"%|"
113 PRINT " |频| +-----+-----+
--+
114 PRINT " | |段|",SPC(8-LEN(STR$(MF*.4-1))),
MF*.4-1,"-";MF*.3;SPC(9-LEN(STR$(MF*.
3))),"|";SPC(4-LEN(STR$(F7))),F7;"人|";SPC
(5-LEN(STR$(FG))),FG;"%|"
115 PRINT " |数| +-----+-----+
--+
116 PRINT " | | |",SPC(8-LEN(STR$(MF*.3-1))),
MF*.3-1,"-";MF*.2;SPC(9-LEN(STR$(MF*.
2))),"|";SPC(4-LEN(STR$(F8))),F8;"人|";SPC
(5-LEN(STR$(FH))),FH;"%|"
117 PRINT " | | +-----+-----+
--+
118 PRINT " | | |",SPC(6);MF*.2-1;"分以下";SPC
(7-LEN(STR$(MF*.2-1))),"|";SPC(4-LEN
(STR$(F9))),F9;"人|";SPC(5-LEN(STR$(FI))),
FI;"%|"
119 PRINT "+-+-+-----+-----+
--+
120 PRINT " | 优分率 |",YF;"%";SPC(13-LEN(STR
$(YF))),"| 低分率 |",DF;"%";SPC(11-LEN(STR
$(DF))),"| "
121 PRINT "+-----+-----+-----+
--+
122 PRINT " |学科标准差|",S;SPC(8-LEN(STR$(S))),
"| 相对标准差 |",CV;"%";SPC(9-LEN(STR
$(CV))),"| "
123 PRINT "+-----+-----+-----+
--+
124 POKE 1915,4;PRINT
125 PRINT TAB(4);"备注:各率均为以全卷 100 分计算,60

```

```

分以上为及格,"
126 PRINT "80 分以上为优分,40 分以下为低分,各分数段
为满分";PRINT"100 分计算逐级递减 10 分."
127 PRINT;PRINT TAB(4);"2. 分数分布直方图";PRINT
128 POKE 1659,0;HOME;HGR2;HCOLOR=3;HPlot 20,
0 TO 20,151
129 HPlot 18,6 TO 20,0 TO 22,6;HPlot 273,149 TO 279,
151 TO 273,153
130 FOR I=1 TO 10:HPlot 22*I+20,148 TO 22*I+20,
151;NEXT I
131 FOR I=1 TO 6:HPlot 20,151-20*I TO 23,151-20
*I;NEXT I
132 VTab 1;HTab 4;PRINT"人数%";VTab 3;FOR I=60
TO 10 STEP-10;PRINT I;NEXT I
133 VTab 9;HTab 33;PRINT"分 0 10 20 30 40 50 60 70 80
90 100";HPlot 20,152 TO 279,152
134 F(1)=FA;F(2)=FB;F(3)=FC;F(4)=FD;F(5)=FE;
F(6)=FF;F(7)=FG;F(8)=FH;F(9)=INT(F9*
1000/N+.5)/10;F(10)=INT(F0*1000/N+.5)/10
135 FOR I=10 TO 1 STEP-1;IF F(I)=0 THEN 137
136 HPlot 22*(10-I)+20,151 TO 22*(10-I)+20,151
-F(I)*2 TO 22*(11-I)+20,151-F(I)*2 TO 22*
(11-I)+20,151
137 NEXT I
138 PRINT CHR$(19)
139 IF O$="Y" THEN TEXT;PRINT D$;"PR#1";POKE
1913,2;PRINT CHR$(17);PRINT D$;"PR#0"
140 PRINT D$;"PR#3";GOTO 61
141 HOME;RR=PEEK(222)
142 IF RR>10 OR RR<5 OR RR=7 THEN 5
143 ON RR-4 GOTO 144,145,146,147,148,149
144 PRINT"磁盘数据不够! 按任意键返回!";CHR$(19);
GOTO 5
145 PRINT"文件找不到! 按任意键返回!";CHR$(19);GO-
TO 5
146 GOTO 142
147 PRINT"磁盘接口错误! 检查完毕后,按任意键再次存
盘!";CHR$(19);GOSUB 37;GOSUB 48;GOSUB 37;
GOSUB 39;GOTO 5
148 GOSUB 48;PRINT"此盘已满! 换盘后,按任意键再次存
盘!";CHR$(19);GOSUB 37;GOSUB 48;GOSUB 37;
GOSUB 39;GOTO 5
149 INPUT"文件加了锁! 需解锁后再次存盘吗?";O$;IF O
$="N" THEN 5
150 PRINT D$;"UNLOCK";T$;GOSUB 37;GOSUB 48;GO-
SUB 37;GOSUB 39;GOTO 5

```

氯化氢制取的动态显示

江西萍乡中学(337000) 刘 萍

在高一化学氯化氢制取中,水吸收 HCL 气体这一动态现象,在实验室里很难实现,化学老师只能通过

讲解来描述这个过程,我编写一个程序来显示这个动态现象,经过讲述和图形显示两结合、在教学取得了良好的效果。

操作过程

输入程序,运行程序,按任意键——加水,按任意键——加氯化氢气体,在显示屏上,可看到水吸收HCL气体的过程,若要结束实验只需按A键即可。

水吸收氯化氢气体的演示程序

```
10 HGR2:HCOLOR=3;J=140
30 HPLOT 80,90 TO 80,190 TO 179,190 TO 179,80 TO
  70,80 TO 80,90;HPLOT 80,140 TO 179,140
80 FOR I=80 TO 124;HPLOT I,J;
  J=J-1;NEXT I
115 J=96
120 FOR I=135 TO 179;HPLOT I,J;J=J+1;NEXT I
160 HPLOT 124,96 TO 124,40 TO 40,40 TO 40,29 TO 136,
  29 TO 136,96
194 GET A$
196 HPLOT 190,160 TO 190,180 TO 194,180 TO 194,160
197 HPLOT 187,163 TO 192,158 TO 198,163
198 HPLOT 220,160 TO 220,180 TO 217,175
199 HPLOT 210,163 TO 215,170 TO 208,178;HPLOT 230,
  163 TO 225,170 TO 232,178
202 FOR I=1 TO 50
203 HPLOT 80,190-I TO 179,190-I;NEXT I
204 GET B$
210 HPLOT 50,10 TO 40,10 TO 40,14 TO 50,14;HPLOT 47,
  7 TO 56,12 TO 47,17
220 HPLOT 60,5 TO 60,20;HPLOT 60,12 TO 70,12;HPLOT
  70,5 TO 70,20
230 HPLOT 85,7 TO 85,5 TO 77,5 TO 75,7 TO 75,17 TO
  77,20 TO 85,20 TO 85,18
240 HPLOT 91,5 TO 91,20;HPLOT 90,5 TO 92,5;HPLOT
  90,20 TO 92,20
290 C=0;D=3;G=2
300 GOSUB 704;GOSUB 805
370 A=82;B=177
380 FOR I=1 TO 30
385 HCOLOR=3
390 HPLOT A,140-I TO B,140-I
395 A=A+1;B=B-1
396 IF (I/2)=INT (I/2) THEN 400
397 HCOLOR=0
398 HPLOT 81,140+(I-1)/2 TO 178,140+(I-1)/2
400 NEXT I
410 C=1;D=4;G=4;GOSUB 704
610 A=111;B=148
615 FOR J=1 TO 30
620 HCOLOR=0
630 HPLOT A,109+J TO B,109+J
640 HCOLOR=6
650 FOR I=A TO B STEP 3;HPLOT I,109+J;NEXT I
660 A=A-1;B=B+1
664 IF (J/2)=INT (J/2) THEN 670
665 HCOLOR=3
```

```
666 HPLOT 81,140+(J-1)/2 TO 178,140+(J-1)/2
670 NEXT J
672 C=0;D=3;G=2
675 GOSUB 704
676 P=PEEK(-16384)
677 IF P=193 THEN TEXT;RUN
680 GOTO 370
704 IF C/2=INT (C/2) THEN E=6;GOTO 710
705 E=2
710 FOR I=39 TO 124
716 IF D=3 THEN F=4;GOTO 718
717 F=3
718 HCOLOR=0
720 FOR J=30 TO 40 STEP F
730 HPLOT I,J;NEXT J
735 HCOLOR=E
736 FOR J=30 TO 40 STEP D
737 HPLOT I,J;NEXT J,I
760 FOR J=29 TO 96 STEP 3
765 HCOLOR=0
766 FOR I=126 TO 134 STEP 2
767 HPLOT I,J;NEXT I
768 HCOLOR=E
769 FOR I=126 TO 134 STEP G
770 HPLOT I,J;NEXT I
780 NEXT J
803 RETURN
805 A=126;B=133
810 FOR J=96 TO 139
820 FOR I=A TO B STEP 3
830 HPLOT I,J
840 NEXT I
850 A=A-1;B=B+1
860 NEXT J
870 RETURN
880 END
```

磁盘加密一法

广东韶关市四中(512000) 黄晓晖

本法运用改变目录磁道号的方法来达到加密的目的(这里设目录磁道号为\$15),所以无法用目录磁道为\$11的标准盘起动。再加上一些其它加密法,那效果更好。用此加密法,只能用较高级具有全盘复制功能(40道)的拷贝盘才能拷贝,而且必须运行本盘,文件才能运行。方法如下:

- (1)运行 DOS 3.3
- (2)键入 CALL-151 (进入监控系统)
- * AEB5:A0✓
- * B3EF:28✓

(特第18页)

第九章 堆栈程序设计

南京大学大气科学系(210008) 朱国江

§ 5. 堆栈程序设计

堆栈是计算机信息处理中的一个重要概念。它实际上是在随机存储器 RAM 中开辟的某个特定的存储区域,用于对一些重要数据的暂存和保护断点地址。这个存储区域有这样的特点:数据可以连续存入,断点地址可以得到保护,不会发生冲掉已存入数据和断点地址的情况,并且严格按照先存入的数据后取出来的规则工作。这种存储结构就叫做堆栈(Stack)。

6502 规定堆栈设置在第 1 页位置上(第 1 页是指地址码高字节为 01 的存储区域),即只能把 \$ 0100—\$ 01FF 这 256 个单元用作堆栈。由于堆栈中的数据取存是按照“先进后出,后进先出”的顺序进行的,因而还必须有一个“指示器”,随时指出栈顶地址的变化,指出栈区中存入的数据已经堆到了哪里,或者说,其作用是指示堆栈中允许存取操作的当前地址。并且必须具有在数据出栈时有自动加 1 的功能,数据进栈时有自动减 1 的功能,这个指示器叫做堆栈指针寄存器,简称栈指示器,记作 S(堆栈指针 Stack Pointer)。它是一个 8 位寄存器,即 S 表示堆栈地址的低 8 位,而高 8 位固定为 \$ 01。

栈的操作只有两种,一是压入(PUSH),一是弹出(POP),每压入(或弹出)一个内容的字节,栈指针 S 就减 1(或加 1),微处理器就是根据 S 指出的位置(栈顶地址)取存数据,保证了按“先进后出”的原则去转子程序,保护断点,恢复断点,保护现场及恢复现场。

顺便说一句,堆栈在使用过程中,栈指针 S 总是指向栈顶的一个空单元,而 S 的初值通常设在栈底位置 \$ 01FF 处。

关于堆栈的几条指令简述如下:

- 累加器进栈指令 PHA—— $A \rightarrow MS, S-1 \rightarrow S$
 - 累加器出栈指令 PLA—— $S+1 \rightarrow S, MS \rightarrow A$
 - 标志寄存器进栈指令 PHP—— $P \rightarrow MS, S-1 \rightarrow S$
 - 标志寄存器出栈指令 PLP—— $S+1 \rightarrow S, MS \rightarrow P$
- 它们各自的操作码分别为:48,68,08,28。

〔例 1〕数据块的倒序传送

例如,将 \$ 6001—\$ 6010 单元的 16 个数据,按相反次序传送到 \$ 7001—\$ 7010 的连续单元中去。程序如 9.1 所示。

程序 9.1:

```
1000-A2 01    LDX # $ 01
1002-A0 10    LDY # $ 10
1004-BD 00 60 LDA $ 6000,X
1007-48      PHA
```

```
1008-E8      INX
1009-88      DEY
100A-D0 F8   BNE $ 1004
100C-A2 01   LDX # $ 01
100E-68      PLA
100F-9D 00 70 STA $ 7000,X
1012-E8      INX
1013-E0 11   CPX # $ 11
1015-D0 F7   BNE $ 100E
1017-60      RTS
```

运行前:

```
* 6001.6010
6001-01 02 03 04 05 06 07
6008- 08 09 0A 0B 0C 0D 0E 0F
6010- 10
运行后:
* 1000G
* 7001.7010
7001-10 0F 0E 0D 0C 0B 0A
7008-09 08 07 06 05 04 03 02
7010-01
```

程序 9.1 的几点说明:

• 初始化条件:变址指针存于 X 寄存器,开始为 01→X,数据长度计数器 Y。

• 这是一个双循环结构的程序设计。第一个循环, \$ 1004—\$ 100B,第二个循环 \$ 100E—\$ 1016。前者取一个数,进栈保存, $X+1 \rightarrow X, Y-1 \rightarrow Y$, 只要 16 个数未取完,就循环上去(\$ 1004),不断取数,不断进栈保存,当全部源数据取完,进栈保存好后,转入第二个循环。后者,把第一个循环在栈中保存的数,逐一取出来,放入目的地址(开始 \$ 7001),用 X 作计数器,并控制循环次数,一直到所有数(\$ 10=16)全部顺次出栈,并在目的地址一一保存后结束。

• 本例是学习堆栈概念存取原则的一个典型实例。若用中华学习机单步执行命令 S,就可以看到数据如何一个一个地进栈,又如何执行后进先出的原则一个一个地出栈的。

• 若改本题为 255 个数据传送,也是非常方便的,您能有最简便的方法吗?事实上,只要将 \$ 1003 单元改为 FF, \$ 1014 单元改为 00,即可。

〔例 2〕N 个数据的交换

例如,将 \$ 6000—600F 单元的内容和 \$ 7000—\$ 700F 单元的内容交换。见程序 9.2。

程序 9.2:

```

1000-A2 00      LDX # $00
1002-BD 00 60   LDA $6000,X
1005-48          PHA
1006-BD 00 70   LDA $7000,X
1009-9D 00 60   STA $6000,X
100C-68          PLA
100D-9D 00 70   STA $7000,X
1010-E8          INX
1011-E0 10      CPX # $10
1013-D0 ED      BNE $1002
1015-60          RTS

```

运行前:

```

* 6000.600F
6000- 50 51 52 53 54 55 56 57
6008- 58 59 5A 5B 5C 5D 5E 5F
* 7000.700F
7000- 12 13 14 15 16 17 18 19
7008- 1A 1B 1C 1D 1E 1F 20 21

```

运行后:

```

* 1000G
* 6000.600F
6000- 12 13 14 15 16 17 18 19
6008- 1A 1B 1C 1D 1E 1F 20 21
* 7000.700F
7000- 50 51 52 53 54 55 56 57
7008- 58 59 5A 5B 5C 5D 5E 5F

```

几点说明:

• 程序 9.2 非常简短,并且只用了一重循环,就可以完成 16 个数据的互换。事实上本程序通用性也好,只要将 CPX # \$10 中的立即数改成任意数 N(N≤FF),即可完成 N 个数据的互换。不仅如此,源地址(本例是 6000)、目的地址(本例是 7000)也可以任意改变,从而可以完成从任何一个区域向另外一个区域的数据交换。

• 程序 9.2 中数据的交换,实际上是借助了一个中间存储区域作为数据交换的暂存空间,而这个暂存空间不是别的,正好是堆栈。其思路是,取一个数,进栈保存,调另一个数,存入原来数的地方(交换),数据出栈,再放数又一次交换,指针加 1,只要未将全部数据交换后,即循环上去,直到结束。

• 请注意,程序 9.2 只能用 PHA,PLA 指令,而不能用 PHP,PLP 指令。为什么?请读者自行分析。

〔例 3〕保护和恢复现场的方法

前面一节在谈到子程序设计时,曾指出当主程序转入子程序时,若主程序中已经用到的寄存器(指 X、Y、A、P)在子程序中也要用到,而且当子程序返回主程序后,主程序还要继续用这些寄存器。这时,为了防止寄存器内容被冲掉,就有一个保护现场和恢复现场的问题。即:在转子程序前应首先把这些寄存器的状态存入堆栈中保存起来(保护现场),然后执行子程序的主体部份,最后再返回;而当返回主程序之前,还要从堆栈中取出它们,以恢复寄存器原内容(恢复现场)。

下面的程序指出了保护现场和恢复现场的方法。

```

PHP ;P 进栈,保留 P
PHA ;A 进栈,保留 A
TXA ;X→A
PHA ;X 进栈,保留 X
TYA ;Y→A
PHA ;Y 进栈,保留 Y
;...
PLA ;最后进栈的 Y 先出栈,送入 A
TAY ;A→Y,恢复 Y
PLA ;X 出栈,X→A
TAX ;A→X,恢复 X
PLA ;A 出栈,恢复 A
PLP ;P 出栈,恢复 P
;...

```

由此程序可以看出:

• 保留现场时各寄存器进栈的顺序和恢复现场时各寄存器出栈的顺序恰好相反。这是因为各寄存器状态在栈中保存又从栈中出来,必须按先进后出的原则处理,这是堆栈的结构决定的。

• 堆栈指令中有 P 和 A 的进栈指令和出栈指令。因此,P 寄存器和 A 累加器需要进栈时直接使用 PHP 和 PHA 指令;而在出栈时也可直接用 PLP 和 PLA 指令。

• 6502 指令中没有 X 和 Y 寄存器的进出栈指令,因此它们的进栈出栈,需要经累加器 A 中转。即:X 和 Y 寄存器要进栈时,必须先用 TXA 或 TYA 指令,把进栈的内容先送到累加器 A 中,然后再使用 PHA 指令使 X 或 Y 的内容进栈;出栈时,则先用 PLA 指令,将 Y 或 X 的内容送入累加器 A 中(出栈),然后再用 TAX 指令恢复 X,或用 TXA 指令恢复 Y。

〔例 4〕标志寄存器 P 进出栈的一个实例

关于求最大值的问题,我们曾在循环程序设计(例 2,程序 7.3)和子程序设计(例 3,程序 8.3)中作过介绍,其中程序 8.3 中就是一个标志寄存器 P 进出栈的实例,不妨将它重新写出,并分析一下它保护的是什么现场,有什么用,恢复的是什么内容,应该在什么时候用。程序 8.3 的主程序:

```

1000-A9 01      LDA # $01
1002-85 06      STA $06
1004-A9 60      LDA # $60
1006-85 07      STA $07
1008-AC 00 60   LDY $6000
100B-20 11 10   JSR $1011
100E-85 08      STA $08
1010-60          RTS

```

前面 4 条指令,是把源数据块首地址送入 \$07 和 \$06 单元,第 5 条指令将数据块长度存入 Y 寄存器,以后是转子程序,待子程序执行完后存结果再结束。

程序 8.3 的子程序:

```

1011-A9 00      LDA # $00
1013-88          DEY
1014-08          PHP
1015-D1 06      CMP ($06),Y

```

```

1017-B0 02 BCS $101B
1019-B1 06 LDA ($06),Y
101B-28 PLP
101C-D0 F5 BNE $1013
101E-60 RTS

```

这段子程序的任务是找最大值。我们主要看 \$1013—\$101D 这段循环,而特别注意循环程序中标志寄存器 P 进栈指令 PHP 和出栈指令 PLP 的安排。循环前 A 中的内容为 00,这是一个假想的最大值(也可以取数据块中的第一个数据)。

执行 DEY 指令使 Y-1→Y,立即执行 PHP 指令,这样就使 P 进栈,从而保存了零标志 Z,如果执行 DEY 指令后结果为 0,则 Z 被置 1;反之 Z 被置为 0。这就为以后数据块长度是否被比较完留下了一个判断标志。当执行 CMP (\$06),Y 指令后,判断下一个数是否大于最大值,若是,则换最大值放入 A,不是则保留最大值不变,但不管是与否则,最后都转向执行 PLP 指令,这条指令是 P 出栈,给出 Z 标志,再用 BNE 判别是否 Z=0,是 0,返主程序(通过 \$101E 的 RTS 指令),非 0 继续转向 \$1013 处循环再比。然后 Y-1→Y,再使 P 进栈,每次均保留 P 的现场,一直到 A 是最大数,使 P 出栈,恢复现场,并当 P 中的 Z 标志位为 1 时,结果为 0(所有数据均比较完),结束子程序操作,返回主程序的断点地址 \$100E,存结果结束。

[例 5] 输出码转换程序

将 6502 的 A 中 8 位二进制数转换成三位 10 进制数,并在显示器上显示(送显示器的字符必须用 ASCII 码)。

```

示范题: a) A=FF (16 进制数)
结果: 显示 255 (10 进制数)
        b) A=80 (16 进制数)
结果: 显示 128 (10 进制数)
        c) A=64 (16 进制数)
结果: 显示 100 (10 进制数)

```

我们先给出完成本题的程序 9.3,然后再给予说明。

```

程序 9.3:
1000-A2 00 LDX # $00
1002-C9 64 CMP # $64
1004-90 06 BCC $100C
1006-E9 64 SBC # $64
1008-E8 INX
1009-4C 02 10 JMP $1002
100C-20 24 10 JSR $1024
100F-A2 00 LDX # $00
1011-C9 0A CMP # $0A
1013-90 06 BCC $101B
1015-E9 0A SBC # $0A
1017-E8 INX
1018-4C 11 10 JMP $1011
101B-20 24 10 JSR $1024
101E-09 B0 ORA # $B0
1020-20 F0 FD JSR $FDF0

```

```

1023-60 RTS
1024-48 PHA
1025-8A TXA
1026-09 B0 ORA # $B0
1028-20 F0 FD JSR $FDF0
102B-68 PLA
102C-60 RTS
102D-A9 FF LDA # $FF
102F-20 00 10 JSR $1000
1032-20 48 F9 JSR $F948
1035-A9 80 LDA # $80
1037-20 00 10 JSR $1000
103A-20 48 F9 JSR $F948
103D-A9 64 LDA # $64
103F-20 00 10 JSR $1000
1042-60 RTS

```

运行: *102DG
显示: 255 128 100
程序 9.3 说明:

• 本程序结构比较复杂,但可以分为三大段: \$1000—\$1023 为子程序 1, \$1024—\$102C 为子程序 2, \$102D—\$1042 为主程序。在主程序中,既有调子程序 1 的操作,又有调监控中子程序(入口地址 \$F948)的操作。而子程序 1 中又包含调子程序 2 的操作,这是一个二重子程序的嵌套结构。而从主程序看,则是一个主程序调三个子程序的复杂操作。

• 主程序中,首先将 A 赋值 FF(即最初输入的 16 进制数),然后转子程序 1,完成 16 进制数到 3 位 10 进制数的转换,结果为 255,再调 \$F948 的监控子程序,它的功能为空一格。然后输入第 2 个 16 进制数 80,转子程序 1,完成 \$80 向 3 位 10 进制的转换,结果为 128,再次调 \$F948 子程序空一格,最后输入第 3 个 16 进制数 64,转子程序 1,得 10 进制数 100,主程序结束。由此可见,主程序是为了不断送数到累加器 A 中,完成 3 个 16 进制数的赋值工作。

• 子程序 1 在执行过程中还要调用子程序 2,从这个角度上说,子程序 1 又相当于是子程序 2 的主程序,这样的主从关系应该明确。这里,分析一下子程序 2,它是在数据换成 ASCII 码之前,首先把累加器 A 的内容(实际上是余数)进栈保护(保护现场),所以返主以后(此时应理解返回子程序 1,子程序 1 为主程序),主程序(即子程序 1)还要再接着使用这个余数,因此, A 中的内容不能由子程序(子程序 2)的执行而丢失。同时,子程序 2 也要用 A 来存放待显示的数,所以就需要先把 A 保护进栈,然后再执行子程序 2 的内容。此外,还应注意,在子程序 2 内容执行完毕返主(子程序 1)之前,必须用出栈指令使恢复 A(余数出栈恢复)的内容(恢复现场),以返主后供主程序(子程序 1)继续使用。

• 作为一般情况,如果主程序和子程序使用的寄存器中,发生冲突的不只是 A,还有 X、Y、P 也发生冲突(转第 31 页)

一九九二年计算机初级软件人员竞赛 试题解答与分析

参考答案

题号	小题号与答案																得分	
一	a	(2)			b	(4)			c	(8)			d	(7)			8	
	e	(10)			f	(12)			g	(13)			h	(16)				
二	1	(4)			2	(2)			3	(3)(6)(8)(5) (1)(2)(7)(4)			4	(4) (6)			22	
	5	(4)(2)(3)			6	(4)(1)(3)(2)			7	(2) (4)			8	(1)				
三	1	B	2	C	3	B	4	B	5	D	6	D	7	A,C	8	A,C	10	
四	1	✓	2	×	3	✓	4	×	5	×	6	×	7	✓,✓	8	×,✓	10	
五	1	B		2	C		3	D		4	A		5	D		15		
六	1	C		2	A		3	B		4	B		5	D		15		
七	(1)	R < B		(2)	R		(3)	R = 0		(4)	B		(5)	R		10		
八	1	D,G			2	A			3	D,A,A			4	F			14	
九	1	A			2	D			3	C,F			4	C,F			26	
	5	C			6	A			7	C			8	B,D,F,G				
十	(1)	屏幕		(2)	诊断程序		(3)	中断		(4)	调试		(5)	联机			10	
	(6)	工作站		(7)	编译		(8)	临时区		(9)	虚拟地址		(10)	缺省				
十一	(1) 程序太大,内存容纳不下.																10	
	(2) 文件分配表坏																	
	(3) 内部栈溢出,系统被停止.																	
	(4) 在拷贝之前,目标文件的内容已经丢失																	
总分																	150	

1991 年计算机软件水平考试试题及解答

本书为全国计算机软件水平考试的复习辅导材料,它是由中国软件行业协会考试指导中心组织清华大学、北京大学、中国科大研究生院、北京信息工程学院、南京东南大学、上海科技大学以及本考试指导中心、北京软件水平考试实施办等单位的有关专家对 1991 年度我国软件水平考试试题(程序员级、高级程序员级、系统分析员级)做了系统分析、研究,并做出了分析和参考答案,以供读者参考,亦可供各部门举办辅导班用作教材。定价 7 元。(邮购时另加 15% 邮费)。邮购地址:北京学院路 29 号,中国软件行业协会高档微机协会(100083)。

试题分析

[第一题] 这道题是数制及其转换的自测题。

表格的第1行:

$$(01000010)_2 = (102)_8 = (42)_{16}$$

$$\text{因此 } c = 102, \quad g = 42$$

表格中的第2行:

$$(127)_{10} = 128 - 1 = 2^7 - 1 = (10000000)_2 - 1 = (01111111)_2$$

$$(01111111)_2 = (177)_8 = (7F)_{16}$$

$$\text{因此 } a = 01111111, \quad d = 177$$

表格中的第3行:

$$(01010101)_2 = 2^6 + 2^4 + 2^2 + 2^0 = 64 + 16 + 4 + 1 = (85)_{10}$$

$$(01010101)_2 = (125)_8 = (55)_{16}$$

$$\text{因此 } c = 85, \quad h = 55$$

表格中的第4行:

$$(143)_8 = (01100011)_2 = (63)_{16}$$

$$(01100011)_2 = 2^6 + 2^5 + 2^1 + 2^0 = 64 + 32 + 2 + 1 = (99)_{10}$$

$$\text{因此 } b = 01100011, \quad f = 99$$

[第二题]

1. 同一个逻辑函数可以有繁简不同的表达式, 实现它的电路也不同, 采用什么样的表达式, 才能使电路所用的元器件最少、设备简单呢? 一般说来, 逻辑表达式越简单, 电路使用的元器件就越少。

常用的化简方法是代数化简法, 直接利用布尔代数的基本公式和规则进行化简。

这里介绍几种常用的化简方法:

(1). 合并项法: 利用公式 $AB + A\bar{B} = A$, 将两项合并为一项。

$$\begin{aligned} \text{例如: } & A(BC + \bar{B}C) + A(\bar{B}C + \bar{B}\bar{C}) \\ &= ABC + A\bar{B}C + A\bar{B}C + A\bar{B}\bar{C} \\ &= AB + A\bar{B} \\ &= A \end{aligned}$$

(2). 吸收法: 利用公式 $A + AB = A$ 及 $AB + \bar{A}C + BC = AB + \bar{A}C$, 消去多余项。

$$\begin{aligned} \text{例如: } & \bar{A}B + A\bar{B}CD(E + F) \\ &= \bar{A}B \end{aligned}$$

(3). 消去法: 利用公式 $A + \bar{A}B = A + B$, 消去多余项。

$$\begin{aligned} \text{例如: } & AB + \bar{A}C + \bar{B}C \\ &= AB + (\bar{A} + \bar{B})C \\ &= AB + \bar{A}\bar{B}C \\ &= AB + C \end{aligned}$$

(4). 配项法: 将某一个乘积项乘以 $(1 = A + \bar{A})$, 这样使一项拆为两项, 或者利用公式 AB

$+ \bar{A}C = AB + \bar{A}C + BC$, 左逻辑表达式中增加 BC 项, 再与其它乘积项合并化简, 达到求得最简单结果的目的。

此题可以用配项法化简。

$$\begin{aligned} & \bar{A}\bar{B} + \bar{B}C + \bar{B}C + \bar{A}B \\ &= \bar{A}\bar{B} + \bar{B}C + \bar{B}C(A + \bar{A}) + \bar{A}B(C + \bar{C}) \quad (\text{配项}) \\ &= \bar{A}\bar{B} + \bar{B}C + \bar{A}\bar{B}C + \bar{A}\bar{B}\bar{C} + \bar{A}BC + \bar{A}B\bar{C} \\ &= \bar{A}\bar{B} + \bar{B}C + \bar{A}C(\bar{B} + B) \quad (\text{合并}) \\ &= \bar{A}\bar{B} + \bar{B}C + \bar{A}C \end{aligned}$$

这道题的答案应选择(4)

在每年的水平考试题中, 都有关于数制及其转换方法、逻辑表达式化简的试题, 在复习时要多做练习题, 以便在考试时能快速、准确地进行计算和化简。

2. DMA 是“直接存储器存取”的缩写。DMA 控制器是控制外部输入输出设备与主存储器直接传送信息(数据)的逻辑电路。

在 CPU 与外部输入输出设备之间传送信息时, 由于 CPU 的速度比输入输出设备快得多, 为了不浪费宝贵的 CPU 时间, CPU 采取分时并行工作的方法。例如, 在磁盘存储器与主存储器之间传送数据期间, CPU 照常执行程序。当磁盘存储器准备好数据时, 向 CPU 发出一个称之为“DMA”的请求信息, 这时 CPU 让出总线, 让出时间为主存储器的一个存取周期, 完成数据写入主存储器的操作, 然后 CPU 继续执行程序。当磁盘存储器再次准备好数据时, 重复上述过程。

这道题的答案应选择(2)。

3. 计算机系统的主存储器用来存放需要立即使用的程序和数据, 它由 CPU 访问, 与 CPU 关系非常密切, 因此要求它存取速度快, 通常由半导体存储器构成。与辅助存储器相比较, 存储容量小、价格高、存储的信息不能永久性地保留。辅助存储器的特点是“存储容量大、价格低、可以永久性地脱机保存信息”。正因为同时使用了这两种类型的存储器, 解决了存取速度、价格成本、存储容量三者之间的矛盾, 提高了计算机系统的性能价格比。

辅助存储器主要有磁表面存储器和光存储器两大类。磁盘、磁带都是磁表面存储器。它的工作原理是将磁性材料涂覆在盘片基体上形成记录介质, 利用磁头与记录介质的相对运动来存取信息。光盘是一种光存储器。它的工作原理是利用激光束在具有感光特性的表面上存储信息。激光

可以聚焦成能量高度集中的极小光点,为高度存储信息提供了可能性。因此,它的存储容量比磁盘大。

4. 一个字节由 8 个二进制位组成。它能表示 $2^8 = 256$ 个字符。由题目的叙述中可知,它已经表示了 26 个英语字母和 10 个数字,因此它还能表示 $256 - (26 + 10) = 220$ 个字符。

二个字节有 16 个二进制位,能表示 2^{16} 个字符;一个字节有 8 个二进制位,能表示 2^8 个字符。前者能表示字符数量是后者的 $2^8 = 256$ 倍。

$$2^{16} / 2^8 = 2^8 = 256$$

这道题的第一个填空应选择答案 (4),第二个填空应选择答案 (6)。

5. 显示设备有若干种分类方法。

按所采用的显示器件分类,有阴极射线管(CRT)显示器、液晶显示器、等离子显示器等。CRT 显示器有黑白和彩色两种;以分辨率的不同,可分为低分辨率显示器和高分辨率显示器两种;以扫描方式的不同,分成光栅扫描显示器和随机扫描显示器两种。

按显示的信息内容分类,有字符显示器、图形显示器和图象显示器三大类。

按显示设备的功能分类,有普通显示器和显示终端两大类。前者的功能简单,只能用于接收视频信号,它的控制逻辑电路和存储逻辑电路都在主机接口板中。后者的功能较多,是一种独立完整的输入输出设备,可以通过标准通信接口接到远离主机的地方使用。

这道题的第一个填空应选择 (4),第二个填空应选择 (2),第三个填空应选择 (3)。

6. 分辨率指的是显示设备所能表示的象素个数。象素越密,分辨率越高,图象越清晰。显示器的分辨率取于显象管荧光粉的粒度、荧光屏的尺寸和 CRT 电子束的聚焦能力。同时刷新存储器要具有与显示象素数相对应的存储空间,用以存储每个象素的信息。

灰度级指的是所显示象素点的亮暗差别,在彩色显示器中则表现为颜色的不同。灰度级越多,图象层次越清楚逼真。灰度级取决于每个象素的刷新存储器单元的位数、CRT 本身的性能。如果用 4 位表示一个象素,则有 16 级灰度或颜色;用 8 位表示一个象素,则有 $2^8 = 256$ 级灰度或颜色。字符显示器只用“0”、“1”两种灰度级,就可以表示字符的有或无,这种只有两级灰度的黑白显示器称为单色显示器;具有多种灰度级的黑白显示器称为多灰度级黑白显示器;具有多种颜色的显示器称为彩色显示器。图象显示器的灰度级

一般为 64 级或 256 级。

为什么要“刷新”?什么是“刷新存储器”?

CRT 器件的发光是由电子束打在荧光粉上引起的。电子束扫描过以后,其发光亮度只能维持短暂的一个瞬间(大约几十毫秒)便消失。为了使眼能看到稳定的图象,就必须在图象消失之前,使电子束不断地重复扫描整个屏幕。这个过程叫做“刷新”。每秒刷新的次数称为“刷新频率”。结合人的视觉生理,刷新频率应大于 30 次/秒,人眼才不会感到闪烁。显示器通常选用电视的标准,每秒刷新 50 帧图象。

为了不断提供刷新图象的信号,必须把图象存储起来,存储图象的存储器称为“刷新存储器”,也可以称“帧存储器”或“视频存储器”。

刷新存储器的存储容量由图象分辨率和灰度能决定。图象的分辨率越高、灰度能越多,刷新存储器的存储容量应该越大。

刷新存储器的存取周期必须满足刷新频率的要求。

这道题的第一个填空应选择答案 (4),第二个填空应选择答案 (1),第三个填空应选择答案 (3),第四个填空应选择答案 (2)。

7. 激光印字机是激光技术和电子照相技术相结合的产物。世界上第一台激光印字机是 1975 年推出的 IBM3800,我国在 1979 年研制成功国产激光印字机。

激光印字机突出速度快,印字质量高,印字分辨率能达到 300 DPI 或 400 DPI (每英寸 400 点)。它是一种击打式硬拷贝输出设备。

这道题的第一个填空应选择答案 (2),第二个填空应选择答案 (4)。

8. 按规定,用电设备交流输入插座是按左零(零线)右火(火线)安装连接的。按这种“左零右火”的规则连接电源线时,能使火线中的电流经过机器内部电源的保险丝管。当供电线路或计算机本身发生过流时,保险丝迅速熔断,保护计算机免受其损害。

在计算机安装完毕后,加电前应用万用表测量供电电压是否和计算机使用的交流电电压数值一致。

计算机随机所带的电源线插头上一般都有明显的标记,火线的标记为 L (Live),零线的标记为 N (Neutral)。

这道题的答案应选择 (1)。

[第三题] 略

[第四题]

1. DOS 命令可分为两大类,即内部命令和

外部命令。

外部命令：指以程序文件的形式存放在磁盘上，使用时必须从存有该命令的磁盘上把文件读入内存后方可执行的命令。而内部命令是指驻留在 DOS 内部的程序，在启动 DOS 时已经调入内存，可以随时被调用执行。

2. 堆栈是一种特殊的线性表，它采用先进后出的原则组织数据。

3. 略

4. 数据库是在文件系统的基础上发展起来的。

5. UNIX 操作系统是一种通用、交互式的分时操作系统。

6. 计算机病毒是由软件引起的。

7. dBASE III 是一种关系型数据库管理系统，用表格数据来表示实体间联系的模型叫关系模型。关系模型是数学化的，它把数据看成二维表中的元素，而这个表就是关系。

8. 反汇编是将机器语言翻译成汇编语言。汇编程序又叫汇编系统，它的功能是把汇编语言编写的源程序翻译成机器语言程序。

[第五题]

1. RND 函数产生的值的范围是 (0, 1)，

INT 函数把给定的参数值取整，

如：INT (5.3) = 5

INT (-5.3) = -6

而 FIX 函数只是截取整数部分如：

FIX (5.3) = 5

FIX (-5.3) = -5

所以 INT (-50+50 * RND (0)) 的取值范围是[-50, -1]，

FIX (-50+50 * RND (1)) 的取值范围是[-49, 0]，

正确答案只能是[-99, -1]。

2. STR\$ 函数是把数字值转化为串，而 LEN 是求串的长度，

LEN (STR\$ (-321)) 的值是4

已知 A\$ = "A123#-3456"

而 MID\$ (A\$, 6, 2) 的意思是从 A\$ 串的第六个字符开始截取 2 个字符，VAL 函数是把数字字符串转换为数值。

所以 VAL (MID\$ (A\$, 6, 2)) 的值是-3，

所以 L=1。

3. 290° 的正弦值是大于-1 小于 0，190° 的余弦值也是大于-1 小于 0，

所以 INT (SIN (290 * 3.14159 / 180)) 的

值只能是-1，

而 SGN (COS (190 * 3.14159 / 180)) 的值也是-1，

所以，正确答案只能是-2。

4. 此图功能是用 X 减去 Y，然后用结果 R 与 Y 比较，如果 R 大于 Y 则把 R 的值赋给 X，然后再减去 Y，这样一直进行下去直到 R 的值小于 Y，这时循环停止，所以这正与求 X 对 Y 的模相一致，所以正确答案为 A。

5. 这三种循环是可以相互转化的，现举一实例说明：

如：10 FOR I=1 TO 5

20 PRINT I

30 NEXT I

40 END

用当型循环可表示如下：

10 I=1

20 WHILE I <= 5

30 PRINT I

40 I=I+1

50 WEND

60 END

用直到型循环可表示如下：

10 I=1

20 PRINT I

30 I=I+1

40 IF I <= 5 THEN 20

50 END

反过来，从当型循环和直到型循环向计数循环转化及当型和直到型循环相互转化也非常容易。

[第六题]

1. 根据定义的函数，第 40 句中的 FNB (FNA (1)) 相当于 FNA (FNA (1)) - 1，

而 FNA (FNA (1)) - 1 = FNA (1²+2 * 1+1) - 1

= FNA (4) - 1

= 4²+2 * 4+1-1

= 24

最后的 Y 值只能是 25。

2. 按照程序中对 A\$, B\$ 赋值，INSTR (4, A\$, B\$) 的值应该是 6。

所以第 40 句中 Y\$ 的值是字符串 B，而函数 LEFT\$ (A\$, I) 是取字符串 A\$ 最左边的 I 个字符，函数 RIGHT 串\$ (A\$, I) 是取字符串 A\$ 最右边的 I 个字符，所以经过第一次循环，I=2，Y\$ = CBBAC，经过第二次循环，

I=1, Y\$=CCBBACC, 最后的答案只能是 CBBACC.

3. 此题应该注意的问题是在 FOR 循环中如果循环体内对循环控制变量不改变, 只有执行 NEXT 语时才改变控制变量, 而在此题中, 在循环体内存在对控制变量的改变语句, 所以, 一定要注意 NEXT 语句对变量的自动改变.

如: I 的初始值是 1, 执行 I=I+4 语句后 I=5, 这时 N=25, 执行 NEXT I 后, I=6, 再执行 I=I+4 时, I=10, 这时 N=25+10²=125, 再执行 NEXT I 后, I=11, 这时 I 的值已超过 10, 退出循环执行 60 号语句, 执行结果是 I=15.

最后结果是 N=125+15²=350.

4. 对于输入的任意一个数 X, $Y=X^2$, 如果 $Y>1$, 那么 $P=P+X=X$, 也就是说 $P=X$ (当 $|X|>1$); 如果 $Y\leq 1$, 那么 $P=P^2=X^2$, 也就是说 $P=X^2$ (当 $|X|\leq 1$),

所以, 这个程序完成的功能是计算下列函数的值

$$P = \begin{cases} X^2 & |X| \leq 1 \\ X & |X| > 1 \end{cases}$$

5. 这个程序的功能是求出 3 到 20 范围的素数.

它的思想是取一个数 M, 这个数不能用 2 到 M-1 范围的整数除, 那这个数就是素数, 只要 2 到 M-1 范围内的整数有一个能整除 M, 那么 M 就不是素数.

[第七题]

1. 80-110 这段程序的功能是求余数.

2. 120-150 这段程序的功能是先判断素数 R 是否为 0, 若不为 0, 那么把新求得的余数赋给 B, 前一个余数赋给 A, 然后转到 80 语句, 再求下一个余数.

3. 如果新的余数为 0, 那么前一个余 B 就是所求的结果.

[第八题] dBASEIII 中用户文件有 9 类, 即数据库文件(.DBF), 记忆文件(.DBT), 索引文件(.NDX), 屏幕格式文件(.FMT), 标签文件(.LBL), 报表格式文件(.FRM), 内存变量文件(.MEM), 命令文件(.PRG), 文本文件(.TXT).

2. Config.sys 是操作系统参数设置文件, 这个文件一般定为:

FILES=20

BUFFERS=24

它表示 dBASEIII 运行时要求 CCDOS 能够同时打开 20 个文件, 缓冲区为 24 个. dBASEIII 技

术指标规定, 能同时打开 15 个文件. 如果 CONFIG.SYS 文件中参数设置不当, 或者启动机器时根本就没有这个文件, 那么操作系统可以打开的文件数目为 8, 而扩充的输入输出操作本身将占用 5 个文件, 留给 dBASEIII 用的文件数只有 3 个, 显然这是不够的, 所以必须通过设置 CONFIG.SYS 改变 DOS 缓冲区数和允许同时打开的文件数的默认值.

另外注意此文件必须放在操作系统盘的根目录上.

3. dBASEIII 有 10 个工作区, 编号 A--J, (1--10), 在每个工作区中可打开一个 .DBF, 一个格式文件, 七个索引文件, 各个工作区互不影响, 刚进入时默认为一号.

4. dBASEIII 的字段类型有五种, 分别是数值型, 日期型, 记忆型, 字符型, 逻辑型.

[第九题]

(1). USE GZ 后文件指针指在第一号记录, 所以 EDIT 编辑 1 号记录.

(2) 打开 GZ 后, GOTO 3 指针定位在 3 号记录, 但最后执行 APPEND BLANK 后指针指到文件尾并且加 1, 为当前指针值, 即 7.

(3) USE GZ

REPLACE 工资 WITH 工资+20 FOR 工龄 > 15

满足条件工龄 > 15 的记录有 3 号, 6 号记录.

(4) 打开 GZ 后, 执行 LOCATE 语句, 找到满足条件: = "高工" 且工龄 > 15 的 3 号记录, 执行 CONTINUE 语句后指针指向下一个满足条件的记录, 即 6 号记录.

(5) 打开 GZ 后, GOTO 3 指针定位于 3 号记录, 执行 COUNT 语句, 它对 3 到 6 号记录中满足工龄 < 20 或姓名 = "严六" 的记录计数, 其中共有 4 号, 5 号, 6 号 3 个记录满足条件.

(6) 打开 GZ 后, 在 1 号到 5 号记录中选取职称 = "工程师" 或工龄大于 10 的人, 拷贝到新文件 GZ1 中, 共有 2 号, 3 号两个记录, 马上打开 GZ1, 对工资项求和, 结果为 320.

(7) 逻辑顺序的含义是按编号后的记录顺序, 排序后编号顺序应为 01, 02, 03, 04, 05, 06, 其对应的记录顺序为 1, 5, 3, 2, 6, 4.

(8) 此程序的功能是对工资大于 150 的人计数, 结果放于 A 中并对其工资求和, 结果放于 C 中. 把工资小于 150 的人计数, 结果放于 B 中并对其工资求和, 结果于 D 中.



单片机实验与 BJS-51 实验教程

北京广播电视大学 李广弟

学习单片机是为了使用单片机,因此单片机这门课程的实践性极强,必须在教学活动中加强实验这一理论联系实际的重要环节。

通过实验可以使学生巩固和深化对课程内容的理解,从而把课堂学习引向深入具体;学到单片机系统电路设计及硬件连接的应用技术;掌握单片机汇编语言程序的编写、调试及运行方法;培养学生的独立分析和解决问题的能力以及严谨的科学作风;为今后单片机的应用开发打下基础。

为此,我们在研制 BJS—51 教学实验系统的同时,即着手实验教程的编写工作,以便于配套使用。

本实验教程共包括两部分。其中第一部分为基础实验,可在本系统的最小配置和基本配置上完成。第二部分为扩展实验,要与专用的扩展板配合实现。

基础实验部分共编写约一百个实验。其中包括: MCS—51 汇编语言程序设计实验、I/O 接口实验、LED 显示实验、时钟实验、音乐与声响实验、顺序控制实验、交通灯控制实验、串行通信实验等共八大类。这些实验复盖了单片机应用技术的主要内容,通过实验使学生在汇编语言程序设计、并串行数据传送、显示、中断、查询、控制等基本技术方面得到全面的训练。

本实验教程有如下特点:

1. 适用范围宽广。目前单片机教学活动分布范围很宽,包括从职业高中到研究生的多种层次;在普通教育之外,还有成人教育,以及在职技术干部的培训等。为了满足大范围多层次的教学实验需要,BJS—51 在硬件方面搞了结构的模块化,允许用户根据需要灵活配置系统;在软件方面则通过增加实验题目的数量,并把实验题目分类排档,使用户可根据不同层次的教学要求从中选作,以求扩大其适用范围。

2. 在实验内容上力求复盖单片机应用技术的各个方面。例如:汇编语言程序设计技术、系统扩展技术、并串行输入输出技术、显示技术、查询技术、中断技术、数据转换技术以及控制技术等。在实验方法和要求上,强调实用性,充分作到理论联系实际,以加强学生动手能力和独立解决问题能力的培养。

3. 采用绕接方法,确保实验电路的连接牢固可靠。经验表明,在传统的面板板使用的插接连线方式,时间长了簧片变松,接触不良,常引起实验电路工作不稳定,造成调试困难。这是实验者和指导者均感头痛的问题。为此,本教学实验系统在连接实验电路时,采用绕接连线方式。由于备有专用的绕线工具,因此使用起来方便快捷,而且连线牢固可靠。从根本上解决了实验中

连线接触不良的问题。

本实验教程可直接作为实验指导书使用,教程中对每一个实验都从题目、目的、电路连接、算法、程序流程、参考程序及思考题等几个方面进行说明。这样既可以减轻教师指导实验的负担,也可以减少学生在准备实验上所花的时间,保证实验得以顺利进行。

对于 40~50 学时的单片机课程,以安排 24 小时实验比较合适。每个实验三小时,共作八个实验。如学生的程度较低,可安排四次汇编语言程序设计实验,然后再安排 I/O 接口、LED 显示、时钟和交通灯控制实验各一次。

典型实验介绍

下面列出两个典型实验,供读者了解 BJS—51 实验教程的实验指导方法。

1. 数的排序实验

汇编语言程序设计是单片机实验的基本内容,因此这部分实验在教程中占较大份量。通过这些实验可以使使学生更加熟悉 MCS—51 的常用指令及寄存器的功能和使用,学会汇编语言程序设计的基本方法,掌握典型程序的算法和技巧,为今后的单片机应用开发打下基础。

下面以单字节无符号数的排序为例,说明有关汇编语言程序设计类实验的基本模式。

①实验题目

假定有八个单字节无符号数存放在 RAM 中以 20H 为首地址的连续单元中,请进行升序排序。

②实验步骤和内容

编写、输入、调试并运行程序。

检查程序运行结果,看开序排序是否正确实现。

③算法说明

数据排序的算法很多,本实验以冒泡法实现。

冒泡法是一种相邻数互换的排序方法,因类似水中汽泡上浮,故称冒泡法。执行时从前向后进行相邻数比较,当发现数的次序与排序要求的递升顺序不符时,就将这两个数进行交换。通过这种相邻数互换方法,使小数向前移,大数向后移。如此从前向后进行一次,就会把最大数排在最后,第二次时,就会把次大数排在倒数第二位置……

例如原始数组为 50,38,7,13,59,44,78,22 第一次冒泡过程是:

50,38,7,13,59,44,78,22	(交换)
38,50,7,13,59,44,78,22	(交换)
38,7,50,13,59,44,78,22	(交换)

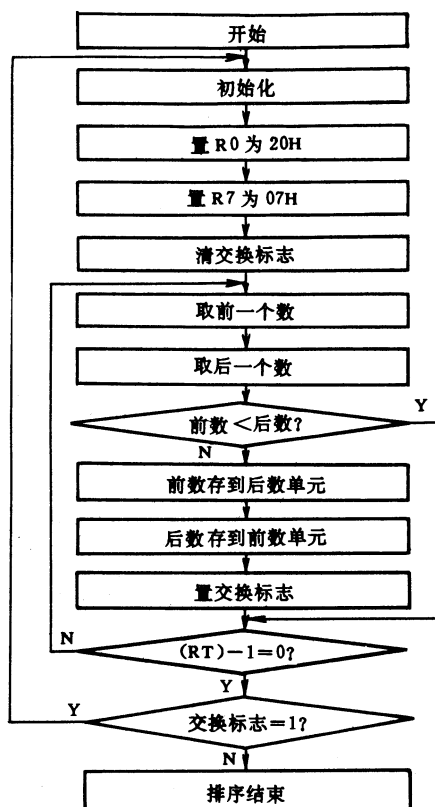
38,7,13,50,59,44,78,22 (不交换)
 38,7,13,50,59,44,78,22 (交换)
 38,7,13,50,44,59,78,22 (不交换)
 38,7,13,50,44,59,22,78 (第一次冒泡结束)

如此进行,各次冒泡的结果是:

第一次 38,7,13,50,44,59,22,78
 第二次 7,13,38,44,50,22,59,78
 第三次 7,13,38,44,22,50,59,78
 第四次 7,13,38,22,44,50,59,78
 第五次 7,13,22,38,44,50,59,78
 第六次 7,13,22,38,44,50,59,78

假定有 n 个数,从上述冒泡过程可以看出,第一次需比较 $(n-1)$ 次,以后各趟的比较次数应该是递减。但为了简化程序,各次一律比较 $(n-1)$ 次。

对于 n 个数,冒泡排序最多需进行 $(n-1)$ 次才能完成排序。但实际上,可能不到 $(n-1)$ 次数即已排好序,如本例共八个数,按说应进行七次,但实际进行到第五次时排序就完成了。判定排序是否完成的最简单方法是看各次排序中是否有数交换发生,如果有数交换,就说明还没排好序,否则就表示排序已经完成。为此,控制排序结束可不使用计数方法而使用交换标志的判定方法。



④参考程序流程

根据上述算法分析,本程序定义如下参数:

R_0 为数据区首地址,初始值为 $R_0=20H$

R_7 为各趟比较次数计数器,初始值为 $R_7=07H$

TR_0 为比较过程中的交换标志

$TR_0=0$ 没交换

$TR_0=1$ 有交换

⑤实验参考程序

8000	7820	SORT	MOV R0, #20H	
8002	7F07		MOV R7, #07H	
8004	C28C		CLR TR0	清交换标志
8006	E6	LOOP:	MOV A, @R0	取前数
8007	F52B		MOV 2BH, A	存前数
8009	08		INC R0	
800A	862A		MOV 2AH, @R0	取后数
800C	C3		CLR C	
800D	96		SUBB A, @R0	
800E	4008		JC NEXT	前数小于后数则转移
8010	A62B		MOV @R0, 2BH	大数放在后边
8012	18		DEC R0	
8013	A62A		MOV @R0, 2AH	二个数交换位置
8015	08		INC R0	
8016	D28C		SETB TR0	
8018	DFEC	NEXT:	DJNZ R7, LOOP	
801A	208CE2		JB TR0, SORT	
801D	020000		AJMP [0000]	返回监控状态

⑥实验参考数据

内存单元 排序前	(20H)	(21H)	(22H)	(23H)	(24H)	(25H)	(26H)	(27H)
排序前	0A	AA	BB	7F	6E	47	64	30
排序后	0A	30	47	64	6E	7F	AA	BB

(未完待续)

(接第 24 页)

突,那么在子程序中还要对 X 、 Y 、 P 的状态加以保护和恢复。因此,转入子程序时应当首先分析一下是否需要保护现场,若需要,那么应注意是哪几个寄存器需要保护,以及在什么时候退回保护。只有这样仔细的加以分析和妥善的处理,才能使程序得以正确执行。本题只有 A 的情况,没有 X 、 Y 、 P 的情况。

关于堆栈程序的设计,我们就介绍到这里。总之,要掌握堆栈的概念、结构、先进后出原则;掌握断点概念、保护和恢复现场的方法;掌握子程序和堆栈之间的关系;掌握 A 、 X 、 Y 、 P 进栈出栈的指令和方法。这是学习本节的要点。



学装微电脑

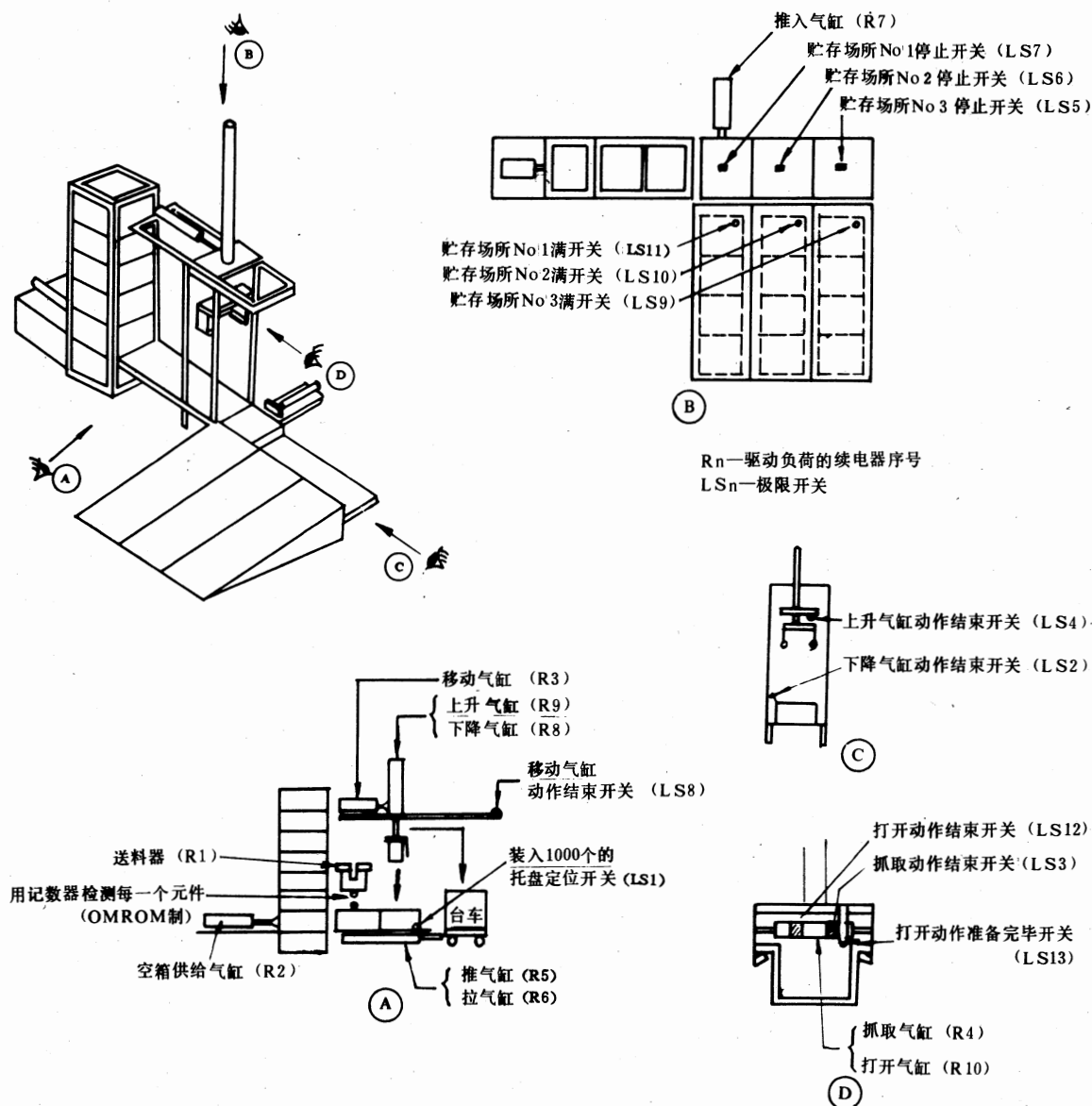
自动输送装置

易齐干

本文介绍冲压件的自动输送装置,讨论极限开关信息输入与控制。冲压件尺寸 $5 \times 15 \times 40\text{mm}$,材料为薄钢板。冲压机一次冲压成型,连续工作,冲压件经过送料器,数量1000个为单位装入成品货架。货架为六层,冲压件被放置在规定的位置。自动输送装置原理如图1所示。工作过程如图2所示。

由图1看出,要输入微电脑14个信息。其中有检测输送装置工作状态的13个极限开关,具体是LS1~LS13。另一个是向货架放置1000个冲压件的计数器。

接收微电脑输出信息的单元有10个。驱动送料器工作的继电器R1,驱动气缸工作的继电器R2~R10共9个。



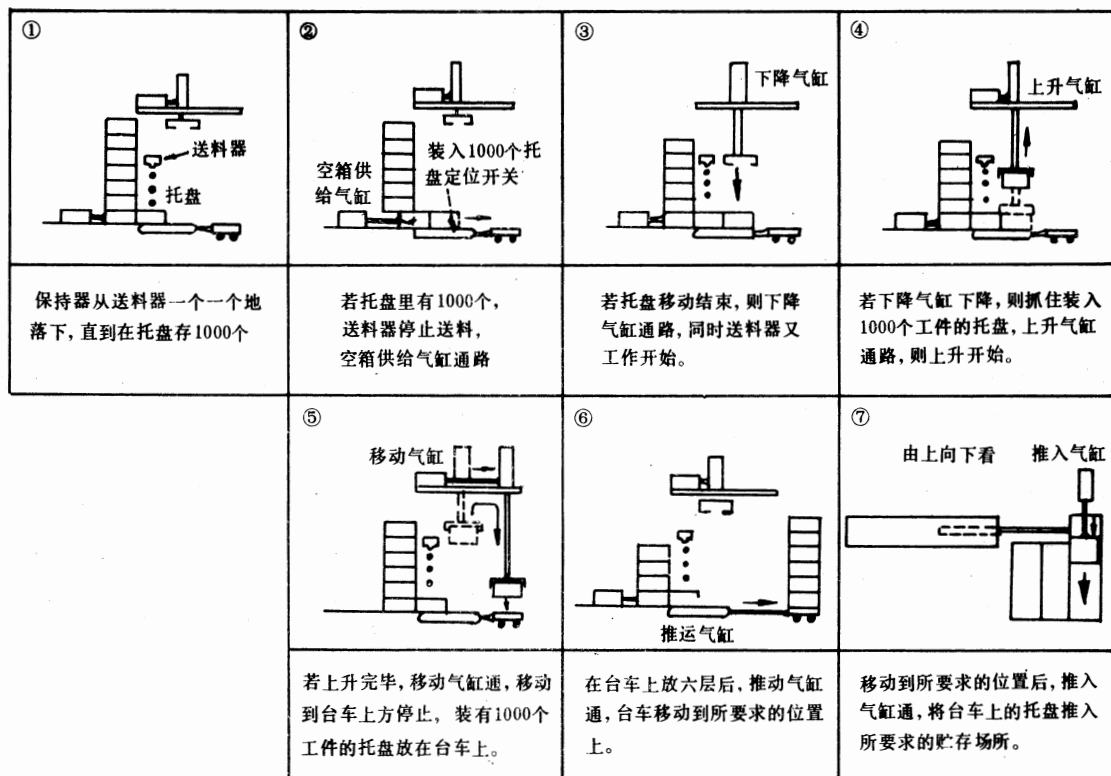


图 2

微电脑输入输出的划分如图 3 所示。

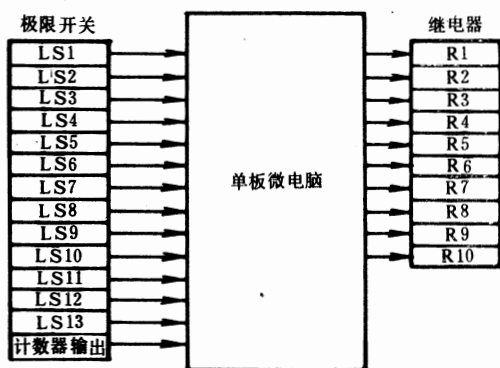


图 3

硬件原理图如图 4 所示。输入给微电脑的信息有前述的 14 个之外, 再加上手动操作 8 个按键开关共计 22 个信息, 采用开关动态扫描方式仅使用 8255A 的端口 C 低 4 位和端口 A (8 位) 合计 12 位即可接收 22 个信息。开关动态扫描方式如图 4, 图 5 为开关动态扫描方式连接。

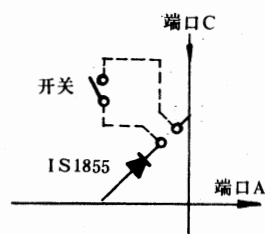


图 5

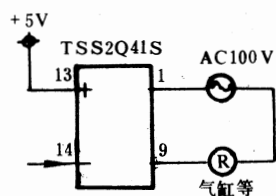


图 6

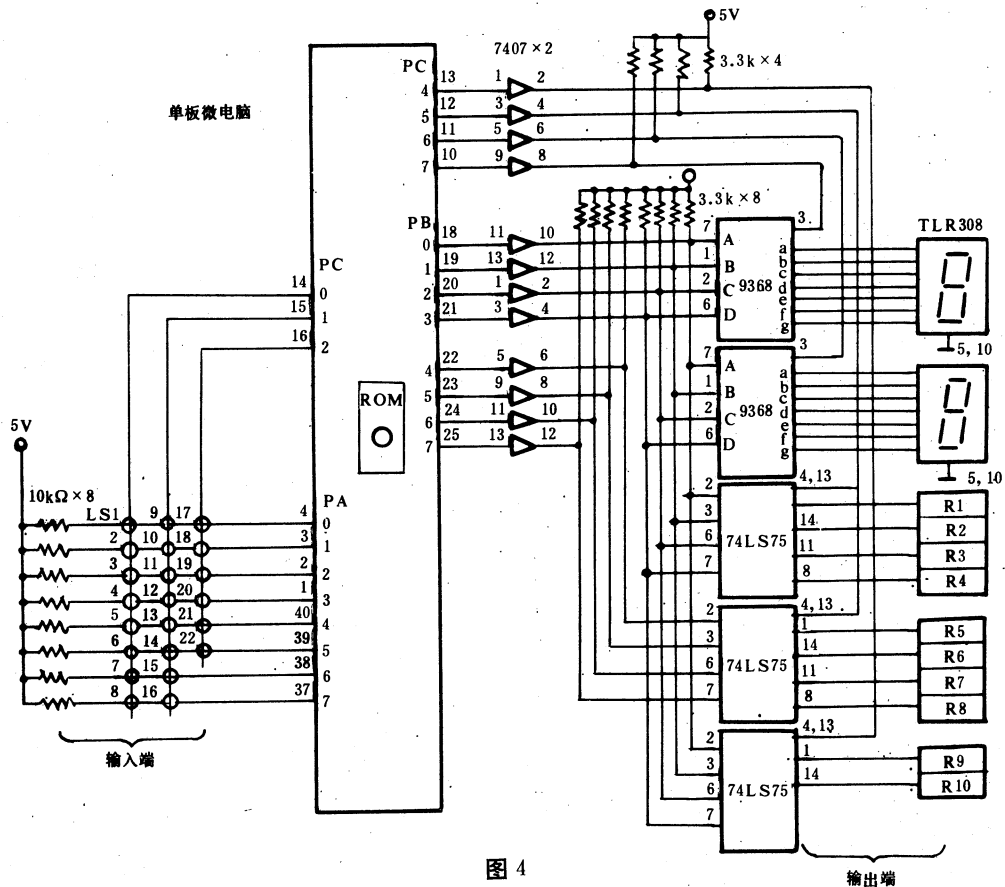
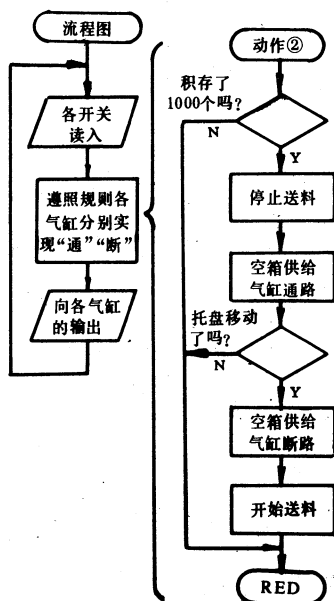


图 4



76543210	助记符	机器语	意义
8000H ① ②	LD IY,8000H	FD 210080	输入数据存放放入IY寄存器。输出数据存放放入HL寄存器。
8010H ③ ④	LD HL,8010H	310180	
计数器的输出信号是高电平吗? (如果计数到1000则输出高电平)	BIT 0,(IY+00H) RET Z	FD CB 0046 C8	计数器的输出不是高电平,不执行下一个指令。
与驱动送料器的继电器相连接的端口其输出为低电平。	RES 0,(HL)	CB 86	8010H的第0位为低电平。
与驱动空箱供给气缸的继电器相连接的端口,其输出为高平。	SET 1,(HL)	CB CE	8010H的第1位为高电平。
装入1000个工件的托盘定位开关(LS1)是通路吗?	BIT 0,(IY+00H) RET Z	FD CB 0146 C8	若8000H的第1位不是高电平,不执行下一个指令。
与驱动空箱供给气缸的继电器相连接的端口,其输出为低平。	RES 1,(HL)	CB 8E	8010H的第1位为低电平。
与驱动送料器的继电器相连接的端口,其输出为高平。	SET 0,(HL)	CB C6	8010H的第0位为1。
进行下一个动作。	RET	C9	

① 托盘定位开关 ② 来自计数器的输入 ③ 继电器驱动供给气缸 ④ 继电器驱动送料器

图 7



图 8 输入输出数据分配

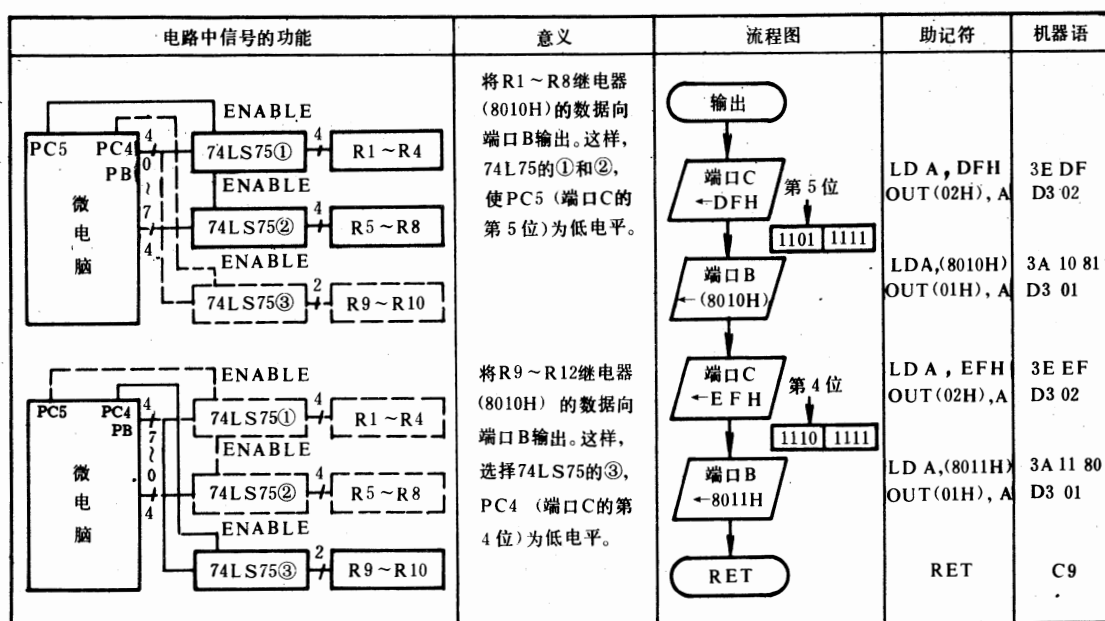


图 9 输出(8010H)、(8011H)的程序

开关动态扫描方式本刊91年曾讨论过,请读者查阅。

微电脑输出信息除前述的10个之外,再加上两个7段数码管。分别显示成品货架数目和放置层数。8255A的端口C高位和端口B共12位为输出端口,显然位数不够。使用74LS75双稳态锁存器扩展输出端口。R1~R10继电器可考虑采用半导体开关,避免产生噪声使微电脑误动作。连接方式如图6所示。9368IC是7段数码管驱动器。7407IC是反相器。

自动输送装置的软件工作方式是前述的极限开关LS和手动按键向微电脑输入ON或OFF信息,与储存在微电脑内的工作程序进行运算比较。输出与工作程序符合的指令,驱动R1~R10继电器工作。

例如,图2中动作②的程序如图7所示。首先,询问计数器“成品有1000个吗?”如果答案“是”,则发出“停止送料”、“供给空箱气缸ON”指令。接着询问“装入1000个工件的托盘定位开关LS1为ON吗?”如果答案“是”,则发出“供给缸OFF”,“送料器为ON”指令。

输入输出的信息预先分配在存储器的8000H~8011H地址中,如图8所示。端口B向继电器R1~R12输出8010H地址8011H地址内容的程序如图9

自动输送装置这个事例,对初学者学习微电脑应用控制很有借鉴,有兴趣的读者不妨将它应用于您的工作中去。



计算机语音输出功能的 开发与应用(上)

马钢南山铁矿技术科(243033) 陈竹林

一个优秀的计算机软件,首先应有一个良好的人机界面,这对方便操作、提高软件的使用效果和软件的生命力是至关重要的。然而,目前人们在人机界面上作的工作大都局限于菜单、图像与音乐的设计上,很少涉及语音的输出。其主要原因是,大多数计算机本身不支持语音输出的功能。由于语音输出在数据及文稿的校对、软件介绍、操作说明和告警等方面有其独特的作用,与显示配合可达到声形并茂的效果,因而越来越为人们所重视。

随着近几年语音处理技术的发展,一些大规模语音处理集成电路相继问世。如 UM5100/5101、T6668 及 TMS5220 等,有些可与计算机接口,对语音信号进行采集、分析、处理与合成,实现高质量的语音输出。采用语音处理芯片研制的语音输出卡也已上市,如中国科学院声学研究所用 TMS5220 芯片开发的语音输出卡,可直接插在 IBM 系列及其兼容计算机内,配以相应的驱动软件,完成标准的汉语语音普通话输出。

虽然语音输出卡能达到很好的语音输出效果,但由于必须增加硬件配置,使应用软件的通用性受到限制,很难在通用性较强的软件(如某些工具软件、管理软件等)上采用。可否在不增加硬件配置的前提下,用软件驱动计算机中的扬声器产生语音输出呢?回答应该是肯定的。问题的关键是如何获得语音数据和如何编制驱动软件。目前人们已经在 APPLE-II 系列计算机上取得了初步的成功。由于这类计算机运行速度慢、内存与外存容量较小,只能完成简单的语音输出,实用价值并不大。本文将从语音信号的采集、语音数据的压缩处理和语音的合成等方面介绍在 IBM 系列及其兼容计算机上开发语音输出功能的方法,并介绍一个简单使用例子,旨在探讨一种简单易行、能被人们接受和采用,在使用广泛的微型计算机上普及语音输出功能的途径和方法。

一、语音采集方案的选择

语音数据是计算机完成语音输出的必要条件之一,通常是通过对人的说话声音进行转换、采集,并经过一定分析处理获得的。声音包含有音调、音色和响度三种特征。其中,音调反映声音的基本频率,在声音波形中,表现为单位时间内过零次数的多少;音色反映声音的高次谐波特性,决定了声音波形的复杂程度;响度反映声音的强弱变化,在声音波形中,表现为幅度(或包络)的变化。虽然三种特征都采集的方案能不失真的合成高质量的声音信号,但采集与合成较为复杂,一般需要增加语音输出的硬件配置,我们不易采用。根据声音信号的特点,忽略音色与响度,只采集

音调信息,合成的语音也能满足人们听懂的要求。采用这一方案可大大简化语音数据采集、数据处理及扬声器驱动软件复杂程度,采集装置也比较简单,使开发与使用的难度大为降低。

二、采集装置与原理

采集装置是一种能将人的话音转换成语音数据送往计算机的硬件设备。由于我们只采集语音信号的过零信息,这种装置十分简单,仅有一个话筒、三只晶体管 and 几个阻容元件组成(如图①所示),完全可以自制。我们选择 RS-232 串行接口作为语音数据进入计算机的通道,并利用两个引脚的高、低电平对外接采集装置供电。具体讲,用 RS-232 的载波检测信号线(CD 脚)作语音数据输入端,用数据终端就绪信号线(DTR 脚)和请求发送信号线(RTS 脚)分别给外接采集装置提供正负电源。采集装置与 RS-232 口的联接见图 1。

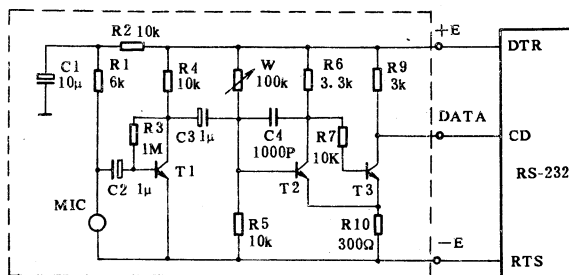


图 1 采集装置及其与计算机的联接

下面介绍其工作原理,用软件向 3FCH 端口(modem 控制寄存器)送 01H 使数据终端就绪端 DTR=1,请求发送端 RTS=0,此时采集装置的 +E 端和 -E 端分别得到约为 +9V 和 -9V 的电源,采集装置开始工作。在无声时,晶体管 T1 无信号输出,T2 与 T3 构成的施密特触发器电路输出低电平(T2 截止,T3 导通,DATA \approx -7V)。当我们对着话筒讲话时,声音经过话筒 MIC 转换成电信号,经晶体管 T1 放大和电容 C3 耦合到施密特触发器的输入端,在其输出端 DATA 输出经过整形的语音数据。从图 2 中我们可以看出,采集装置输出的 DATA 只反映了语音信号的过零特征,没有音色变化,幅度也是固定不变的。最后只要用软件循环不断查询 3FEH 端口的 D7 位,并记录其变化情况,即可完成采集过程,得到相应的语音数据。查询的速率应大于最高语音频率的两倍,语音信号的频率一般在 300~3400Hz 之间,因此采样速率最好不要小于



范围 $\Delta F_h = F_c/16 \sim F_c/2$, 单位时间占用的字节数 $N_h = F_c/7$ 。低频字节对应的话音频率范围 $\Delta F_l = F_c/128 \sim F_c/16$, 单位时间占用的字节数 $N_l = 2F$ 。

例如, 当采样频率 $F_c = 8\text{KHz}$ 时, 可采话音最高频率 $F_m = 4\text{KHz}$ 。高频字节对应话音频率范围 $\Delta N_h = 0.5 \sim 4\text{KHz}$, 单位时间占用字节数 $H_h = 1.14\text{KB/S}$ 。低频字节对应话音的频率范围是 $\Delta F_l = 62.5 \sim 500\text{Hz}$, 单位时间占用字节数 $N_l = 125 \sim 1000\text{B/S}$ 。存储一分钟的

语音数据, 全部是高频字节时, 占用 68KB 的字节空间, 全部是低频字节时, 占用的字节数随话音频率的降低而减少, 最多时占用 60KB , 最少时占用 7.5KB 。也就是说, 话音频率越低, 数据的压缩效果越明显。尤其在话音停顿期间, 采集装置的输出 DATA 恒为 0 , 数据压缩比最高。

(待续)

怎样使 CCDOS2.10 的 9 针打印驱动程序适应 2.13H 汉字系统

湖南岳阳市 6906 工厂 (414007) 李修连

2.13H 有丰富的汉字处理功能, 但它没有提供 9 针 (包括 12 针) 打印驱动程序。而目前我国仍有一些用户使用 9 针打印机。还有些用户有可利用的 9 针打印机。为了能够把这些 9 针打印机利用起来, 我们修改 CCDOS2.10 的 9 针打印驱动程序 ALL9P.EXE 和 NEW9P.EXE, 使之适应 2.13H 汉字系统, 取得了很好的效果。

ALL9P.EXE 中, 从 $1\text{FD}0\text{H}$ (在 DEBUG 下) 开始的三条指令取得汉字字模的段址, 入口 AX 中存放汉字机内码, 出口 DX 中存放字模的段址 (偏移量为 0)。

2.13H 汉字系统中由中断 7F 取得汉字字模的段址, 入口参数 DX 为汉字机内码, 出口参数 DX 为字模的段址。因此我们只要修改 ALL9P.EXE 中取汉字字模段址, 三条指令就可以使它适用于 2.13H 汉字系统

修改过程如下:

```
A>REN ALL9P.EXE 9P
A>DEBUG 9P
-A1FD0
```

```
MOV DX,AX
INT 7F
NOP
NOP
NOP
NOP
NOP
-W
-Q
A>REN 9P ALL9P213.EXE
```

NEW9P.EXE 中, 从 $1\text{FE}0\text{H}$ 开始的三条指令取汉字字模的段址, 其入口参数、出口参数同 ALL9P.EXE 一样。因此, 除地址不同外, 修改 NEW9P.EXE 的方法与修改 ALL9P.EXE 基本相同。

修改后的驱动程序可直接安装, 不需运行 FILE16B.COM 。

注: 本稿是用修改后的 NEW9P.EXE 在 2.13H 下打印的。

8031 真的无 ROM 吗?

北京理工大学三系 90 研 (100081) 肖革文

现在出版的许多单片机书籍中都说 MCS-51 系列中的 8031 内部无 ROM, 因而必须外接 EPROM, 实际上, 8031 内是有 ROM 的, 而且 ROM 中有内容。这一事实, 作者在江苏启东计算机厂生产的 DVCC-51 开发机上, 读出了 8031 的内容, 并在终端屏幕上显示程序机器码, 且可反汇编显示, 亦可用打印机把屏幕内容打印出来。

读出 8031 内容的方法:

在 DVCC-51-ED 开发机上, 将 8031 芯片插在

EPROM 固化区 8751 的位置上, 用键盘键入 $0000\text{ F1 } 1000\text{ F20 EPMOV}$, 即把 8031 内容读到 000H 内存。然后用反汇编命令显示或打印, 并可通过 IBM-PC 及其兼容机存盘。

由此可见, 8031 内部是有 ROM 的, 只不过是用户不能使用而已, 其容量也为 4KB 。由此推论: 8032, 8344, 8035, 8039, 8096 内也有 ROM, 且 ROM 中也有内容, 通过分析其中的内容, 我们可以学到国外编程的技巧, 也可以利用其中的一些子程序为我所用。



IBM-PC/XT 及其兼容机 RAM 故障的检修方法

黑龙江绥化师范专科学校(152061) 齐吉泰

IBM-PC/XT 及其兼容机的随机存储器 RAM 芯片损坏所出现的故障现象主要有下面三种:

- 1、自检过程中显示错误代码。
- 2、自检过程中不显示错误代码,也能够引导操作系统,但所显示的存储容量与实际不符。
- 3、开机后无法完成自检和系统初始化,不显示光标和任何信息。

对于上述由 RAM 芯片损坏而出现的故障,可根据故障现象分别采用下面的方法进行检修。

一、显示错误代码时 RAM 故障的检修

多数 PC/XT 及其兼容机,除第一排 RAM 外,其它各排中的任意一只 RAM 芯片损坏时,机器在自检过程中都会给出如下信息:

```

XXX      KB      OK
XXXXXX   XX      201
ERROR    (RESUME=F1 KEY)
  
```

其中第一行数字给出了通过自检的内存容量;第二行的第一位数字表示损坏的 RAM 芯片所在的 BANK,即某排 RAM;第六、七两位数字表示损坏的 RAM 芯片在 BANK 中的 Bit,即具体位置;201 表示故障在 RAM 电路中。

PC/XT 及其兼容机的随机存储器 RAM 由于内存容量的差异和采用不同的 RAM 芯片,使其在机器中的位置和排列次序也不尽相同。现以扩充到 640KB 的机型为例,如果内存全部采用 64K×1 位的 RAM 芯片,则系统板上安装的存储容量为 256KB,分为四个模块,如图 1 所示。剩下的 384KB 安装在内存扩充板上,分为六个模块,如图 2 所示。640KB 的存储容量共分为十个模块(BANK),每个模块共有九块 RAM 芯片,各模块所对应的错误代码见表一,每个模块中的各个 RAM 芯片所对应的错误代码见表二。明确了这个排位之后,根据屏幕上显示的错误代码,就可以很快地知道损坏的 RAM 芯片在哪个模块中的哪一位。例如自检

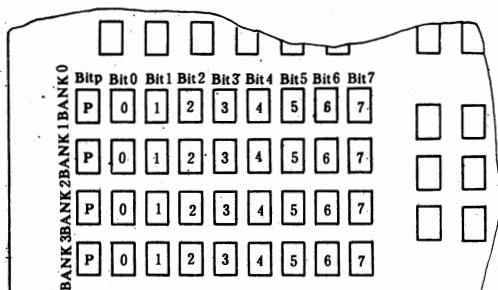


图 1 系统板

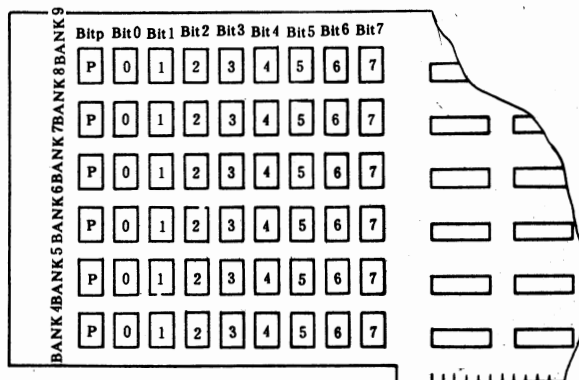


图 2 扩充板

表一

板 名	模 块	错误代码
系统板	BANK 0	00000
	BANK 1	10000
	BANK 2	20000
	BANK 3	30000
扩充板	BANK 4	40000
	BANK 5	50000
	BANK 6	60000
	BANK 7	70000
	BANK 8	80000
	BANK 9	90000

表二

位 置	错 误 代 码
Bit p	00
Bit 0	01
Bit 1	02
Bit 2	04
Bit 3	08
Bit 4	10
Bit 5	20
Bit 6	40
Bit 7	80

时显示的错误代码是 10000 00 201,则由 201 可以断定故障存 RAM 电路中,由 10000 可以断定故障存系统板上的 BANK1 中,由 00 可以断定故障在 BANK1 中的 BitP,即第二排 RAM 中的第一只 RAM 芯片。如果自检时显示的错误代码是 90000 80 201,则查表可知损坏的 RAM 芯片在扩充板上 BANK9 中的 Bit7。

如果内存采用的是 256×1 位和 64K×1 位两种 RAM 芯片,则 640KB 的存储容量全部安装存系统板上,也分为四个模块,排列次序和图 1 所示的相同,所不同的是前两排采用了 256×1 位的 RAM 芯片,使每排的存储容量增大到 256KB,后两排仍然使用 64K×1 位的 RAM 芯片,每排的存储容量还是 64KB。这种机型内存的各模块所对应的错误代码见表三。

表三

模 块	BANK0	BANK1	BANK2	BANK3
错误代码	00000	40000	80000	90000

每个模块中的各个 RAM 芯片对应的错误代码和表二所列的相同。如果自检时显示的错误代码为 40000 00 201, 则损坏的 RAM 芯片的位置在 BANK1 中的 BitP, 即第二排的第一位 RAM 芯片; 如果出现错误代码为 90000 80 201, 则损坏的 RAM 芯片的位置在 BANK3 中的 Bit7。

根据上述方法一般都能比较准确地确定出损坏的 RAM 芯片的位置。更换 RAM 芯片, 故障即可排除。对于其它类型的微机。根据显示的代码和 RAM 芯片的容量及排列情况, 参照上述方法也能够比较准确地确定出损坏的 RAM 芯片的位置。

二、不显示错误代码时 RAM 故障的检修

有一些 PC/XT 及其兼容机, 当随机存储器中某一 RAM 芯片损坏时, 只要损坏的芯片的位置不在 BANK0 或其它各模块的 P 位, 则开机后能够自检和初始化, 虽然自检时屏幕上显示的存储容量与实际安装的不符, 但不显示错误代码。例如, 一台内存为 640KB 的 PC/XT 微机, 开机后自检到 512KB 时转去引导操作系统, 内存“丢失”128KB, 这种内存“丢失”的故障, 由于不显示错误代码, 一般无法直接确定损坏的 RAM 芯片所在的 BANK 和 Bit。对于这种类型的微机, 如果有高级诊断盘, 可以利用诊断程序确定损坏的 RAM 芯片的位置。利用高级诊断盘诊断 RAM 故障时, 显示的错误代码和自检时出现的错误代码是相同的, 因此确定损坏芯片位置的方法也相同。高级诊断盘的使用方法, 有关资料已有详细的介绍, 这里不再赘述。

在没有高级诊断盘的情况下, 为了避免盲目地更换 RAM 芯片, 首先应确定损坏的 RAM 芯片所在的 BANK。确定故障所在的 BANK, 可以根据自检时屏幕上显示通过的内存容量和所使用 RAM 芯片的存储容量来确定。例如, 通过自检的内存容量为 512KB, 对于全部使用 64K×1 位 RAM 芯片的微机, 因为每个模块的存储容量都是 64KB, 自检通过了 512KB/64KB=8 个模块, 故此可以断定故障在扩充板上第九个模块, 即 BANK8。而对于使用 256K×1 位和 64K×1 位两种芯片的微机, 因前两个模块中使用的是 256K×1 位的芯片, 则自检通过了 512KB/256KB=2 个模块, 故此可以断定故障在第三个模块中, 即 BANK2。

由此可见, 当通过自检的存储容量中含有 N 个模块时, 故障在 N+1 个模块中。

在确定出损坏的 RAM 芯片所在的 BANK 后, 逐个或全部更换该模块的 RAM 芯片, 故障即可排除。这样虽然不能直接确定损坏芯片的准确位置, 但可以缩小故障范围, 避免逐行更换。

三、第一排(BANK0)RAM 故障的检修

开机后无法完成自检和系统初始化, 不显示光标和任何信息, 使整个系统“挂起”的故障现象多为第一排(BANK0)RAM 损坏而造成的, 由于屏幕上不显示任

何信息, 因此无法确定损坏的 RAM 芯片的位置。虽然可以将第一排 RAM 芯片个更换, 但多数 PC/XT 及其兼容机的第一排 RAM 芯片是直接焊在印刷电路板上的, 更换时要一只一只地烫下来试探着更换。为了找到一只损坏的芯片要烫下好几只, 这样即费工又容易损坏印刷线路板。下面介绍一种既能判断上述故障是否在 RAM 又能给出第一排(BANK0)中损坏芯片位置的检修方法。

由于系统板上四排 RAM 的数据线和地址线都是公用“总线”, 各排 RAM 所不同的只是 $\overline{\text{RAS}}$ 和 $\overline{\text{CAS}}$ 信号。第一排(BANK0)使用的是 $\overline{\text{RAS0}}$ 、 $\overline{\text{CAS0}}$ 信号, 第二排(BANK1)使用的是 $\overline{\text{RAS1}}$ 、 $\overline{\text{CAS1}}$ 信号, 其余类推。 $\overline{\text{RAS}}$ 和 $\overline{\text{CAS}}$ 信号是由系统板上的 U42、U56 (不同的 PC/XT 机标号有所不同) 两个三-八译码器译码后提供的, 如果将 BANK0 和 BANK1 的 $\overline{\text{RAS}}$ 、 $\overline{\text{CAS}}$ 信号对调, 也就是将第二排 RAM 变成 BANK0, 将第一排 RAM 变成 BANK1。如果第二排 RAM 完好, 对调后系统能够完成初步自检并出现光标和显示错误代码, 这不但说明了故障在第一排 RAM, 而且损坏芯片的位置也由错误代码给出。根据前面介绍的利用错误代码确定损坏的 RAM 芯片的方法, 找出损坏的 RAM 芯片的位置, 将该片烫下来, 换上同型号的新芯片, 故障即可排除, 这样就避免了一只只焊下来更换。

具体做法是: 在系统板上找到 74LS138 译码器 U42、U56, 把 U42 第 14、15 脚的引线用锋利的小刀割断后, 用细导线互相对调连接; 把 U56 的第 10、11 脚引线也割断并互相对调连接, 见图 3。

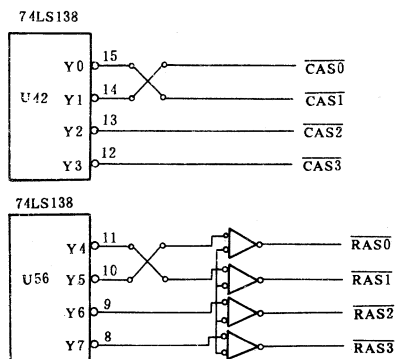


图 3 译码电路

焊好后要用万用表的电阻档测量一下各连线是否可靠, 有无碰线、短路等, 确定无误后方可开机, 如果电路中仅是原来的第一排的某个 RAM 芯片损坏, 则稍候即可显示光标和错误代码 (此时屏幕上显示的错误代码是 BANK1 的信息, 实际查找时应到 BANK0), 损坏的芯片更换后, 将 U42、U56 对调的引线复原。

在维修实践中, 利用上述检修方法, 对于由 RAM 芯片损坏而引起的故障现象, 基本上都能够准确地诊断并加以排除, 但其它因素如线路板断裂, 扩充板开关位置不对等也会造成类似的故障现象, 因此在维修过程中也必须加以考虑。



第二讲 故事情节和游戏结构

山东苍山县机械电子化学工业局(277700) 于 春

电脑游戏一般分为战争、冒险、管理、智能四大类。不管那一类游戏都有一个中心内容,能够完全细致地描述一个事件的发生、发展、结束全过程;有一定的知识性、启迪性和趣味性;有一定的难度和技巧,以激发游戏者的好胜心和竞争意识,促使他们思索、推理、判断、总结,从而达到游戏中增长知识,娱乐中提高才干的目的。

游戏程序是如何编制的呢?它一般要经过确定故事情节、规划游戏结构、确定程序结构、绘制程序框图、编写详细程序五个步骤。如果把游戏比作机械产品,故事情节就是原材料,游戏结构是加工图纸,程序结构是零件毛坯,程序框图是工艺文件,编制程序就是产品的最后加工。

众所周知,图纸是机械制造的依据。游戏结构则是编制游戏程序的关键,是故事情节的再创作、深加工。那么游戏结构的内容如何呢?欲说清这一问题,还须从故事情节说起。

我们知道许多战争、冒险类游戏都是从一个特定的故事情节开始的。如最为游戏者喜爱的双打游戏“魂斗罗”的故事情节是这样的:“公元 2631 年,世界大同,进入了和平时期。但是,地球海军陆战队司令部在一项绝密调查中发现了异星侵略者消灭人类的计划。海军陆战队所属上等兵比尔·拉依乍和兰士·比恩受命歼灭异星侵略者。于是二人便向已变成侵略者要塞的嘎尔嘎戈岛进军。他们要突破原始森林、外围基地、瀑布天险、防御基地、雪山屏障、动力中心、飞机仓库七个关口,最后抵达外星人的栖息处,击毙外星人。摧毁敌基地后,乘飞机胜利返航。”

故事情节犹如文学创作的素材,有了故事情节,才能决定游戏结构。如魂斗罗的游戏结构是这样的:“这是一个战争游戏。因此首先给主人翁配备了光电子弹、火焰喷射器、火球、流星、辐射弹等各种武器。其次由于要经过各种复杂地带,主人翁要具有潜水、跳跃、飞腾的本领。第三,要设计各种不同的游戏画面、音乐以及与其对抗的敌人。第四,为增加游戏难度,把游戏分成了八大关,每关的关头都要与一个势力较强的敌人搏斗。若战术不力,很难过关……”等等。

归纳起来,所有游戏的游戏结构其本质上都是相同的。它一般包含以下几项内容:

1. 游戏的背景画面

游戏的背景、画面如同影剧中的布景,要根据游戏的故事情节,决定画面的种类、色彩、数量、变化方式和复杂程度。以增强游戏的艺术效果,使游戏更形象,更生动,更富有吸引力。

2. 游戏的音响效果

为使游戏逼真,一般游戏中都配有音响效果,如开枪、炸弹爆炸、敌人被击中,建筑物被摧毁等情节,都要有特定的音响,以烘托游戏的气氛。再如飞机、汽车的引擎声,人和动物行走的脚步声等。以增加游戏的真实感。精彩的游戏与丰富的音响是密切相关的。

3. 文字说明

文字说明,一般包括游戏的故事梗概、玩法说明、人物对话、积分、奖励、武器数量等方面的叙述和表达,以弥补画面显示的不足。

4. 游戏者的动作

游戏者指游戏中的主人翁。要规定主人翁的动作、本领,如跑、跳、潜水、腾空、射击、出拳、飞脚等。主人翁每一次动作后,都要计算其动作后的结果。若主人翁已取胜或失败,则进行结束处理(或者接关继续、或者重新开始)。否则,主人翁或另一位对手(或电脑)作另一次动作。动作的计算是游戏程序的核心。

5. 游戏的选项

游戏中可选项的类型、选择的方法和变量的修改,要根据选中的故事情节确定。如难度、关卡、武器、人数的选择等。而且一个选项可衍生出多层选项。如我们选中攻击,则继后就要有射击、出拳或踢脚等动作功能的选项。若选中射击,则又有各种武器的选项等等。选项一般与文字说明共存,也可以用画面选项。每个游戏选项的多少、选项的层次与游戏结构有关,同时还受电脑内存的约束。

6. 游戏的难度

一个有趣的游戏既要容易理解,又要有足够的难度;既要容易入门,又要在短时间内很难玩好。一般是设计许多关,开始几关较容易攻破,以后难度越来越大,要求技巧也越来越高。不动一番脑筋,不下一番功夫很难破关。只有这样,才能吸引游戏者不断思索,探讨,津津有味地玩下去。如多版游戏“淘金者”,I、II代各有 50 关。第 II 代的最后一关,难度最大,很多游戏者都过不去,必须精密策划,合理安排,快速行动才能过关。因此它是一个智能和技巧密切结合的典型游戏,受到广大游戏者的青睐。不难想象,一个很容易打穿的游戏,肯定令人乏味,难以引起兴趣。

游戏结构可粗略地划分以上六点。但并非所有的游戏结构都要具备这六点。如智能、管理游戏,只要有 1、2、3、5、6 几项就可以了。甚至可以更少,只有 3、6 两项也能构成游戏。可以断言,不具备以上几点内容的程序,肯定不是游戏程序,即使勉强凑成,也不会引起人们的兴趣。

另外,在设计游戏结构时,还必须兼顾电脑系统可供使用的内存裕量。诸如魂斗罗类的精彩游戏,结构相

当复杂,背景画面很多,程序极为庞大,一般都用机器语言编写,占用内存达 256K 之多。我们的学习机一般只有 4~8K 的内存供用户使用,所以,我们只能编一些简单的游戏程序。另外,由于用 BASIC 语言编程,程序的执行速度较慢,所以游戏中主人公的动作是较慢的。虽然如此,游戏结构也是必不可少的。编程前,要对游戏程序精心策划,以使较少的内存发挥较大的作用,利用有限的内存空间编制出富有趣味的游戏程序。

现在,我们构思这样一个游戏,名字叫作“成龙救金凤”。故事情节如下:“成龙和金凤是一对热恋的情侣。山妖发现金凤美貌,就派山鹰叼走了金凤。欲打败山妖,救出金凤,必须找到制伏山妖的宝贝——降魔镖。但是,降魔镖是山神的镇山之宝,从不借人,只有觅机盗出。降魔镖秘藏在聚宝楼内,并由两名妖怪巡回看守,而聚宝楼是全封闭的楼房,每月只开一个时辰通风。成龙只有在通风时进入楼内,躲过妖怪,寻到宝贝,离开聚宝楼。否则,将葬身楼内。若被妖怪抓住则立即毙命。勇敢的成龙,机智地躲过妖怪,终于寻到宝贝打死山妖,救出爱侣。”根据故事情节,规化游戏结构如下:

1. 游戏画面

配合游戏结构共设计四幅画面:

a. 开始画面:成龙和金凤正在郊外散步;突然降下山鹰,叼走了金凤;成龙狂奔追赶。

b. 主题画面:七层楼房;金凤显示在楼顶右上角;成龙在楼底左下角;两个妖怪随机出现在 3~7 层楼之间;九个大小、形状不同的宝箱散落在七层楼房的不同角落。

c. 结束画面两幅:一幅是成龙金凤二人欢呼相奔、拥抱。一幅是成龙和金凤手拉手高高兴兴回家。

2. 游戏的音响

a. 游戏开始有一段悠扬的乐曲,伴随着成龙金凤愉快的脚步。突然天空变暗,山鹰飞降,叼走金凤,乐曲紧促、强烈。

b. 成龙救出金凤,相向飞奔,热烈拥抱,有一段激动欢快的音乐。

c. 成龙与金凤手拉手回家时,又是一段悠扬的音乐。

d. 游戏进行中,穿插一些象声音响,如脚步声,上、下楼梯声,打开箱子声等等。

3. 文字说明

由于游戏较简单,不作说明,只打印题头、时间、得分等。

4. 动作

该游戏的主要情节是成龙依次打开箱子,寻找宝贝。因此,成龙的动作仅设计走路,上、下楼梯简单的动作。可用左、右键控制成龙左、右走;用上、下键控制成龙上、下楼;用 START 键控制游戏开始。

5. 游戏的难度

a. 从 27 个图形中随机选取 9 个作为存宝的箱子,箱子有一定的顺序(电脑约定,不标出),成龙只有按顺

序打开第一个箱子,然后再打开第二个、第三个……直到第九个。这犹如宝贝放在第九号箱子里,九号箱的钥匙放在八号箱,而八号箱的钥匙又放在七号箱……。所以必须从一号箱开始,依次打开,才能最终找到宝贝。

b. 两个妖怪在七层楼之间穿梭巡逻,并自动向成龙靠近。

c. 加进时间限制,超过规定时间,没找到宝贝,则游戏结束。

为压缩程序量,省去了选项。欲增加游戏难度,读者可调整时间参数,可把 TI 从 800 调到 600、400。

这就是成龙救金凤的游戏结构,规划好游戏结构后,就可进入下一步骤了。留待下一讲讨论。下面介绍一个简单的赌博游戏程序。

例三 赌博游戏——石头剪子纸

石头、剪子、纸的游戏(也叫剪子、包袱、锤)是小朋友最常玩的赌胜游戏。游戏中,双方在石头(ROCK)、剪刀(SCISSORS)、纸(TISSUE)三者之间任选其一。规则是,石头击败剪刀、剪刀击败纸、纸又击败石头。为了取胜,游戏者必须仔细研究对方的战术,随时调整自己的选择击败对方。游戏虽然简单却饶有趣味,以致成人也常用这种方法决定某些事件的输赢。这个游戏我们也可以与电脑一起玩。中华学习机已有成熟的程序,现改造移植到学习机中。

程序清单如下:

```
5 REM "GAMBLE GAME"
10 CLE.:CLS:CG.0,0
20 LOC.7,3:P."* GAMBLE GAME *"
30 LOC.0,9:P."GUESS ROCK,SCISSORS,TISSUE":P.
40 P."BY PRESSING (R),(S),(T)":P.
50 P."TO END GAME PRESS (ESC)":P.:LOC.10,18:P.
   "1992.3."
60 GOS.360:CLS:GOS.310
70 PLA.B 0,RND(12)+1,13,29,45
75 LOC.0,0:P."YOUR TOTAL":T(2)
80 LOC.15,0:P."MY TOTAL":T(1)
90 LOC.6,8:P."":LOC.18,8:P."
   ":LOC.10,15:P."
100 M=1+ABS(N(S,1))<=N(S,2))
110 M=M+ABS(N(S,M))<=N(S,3))* (3-M)
120 R=S:M=M+3*ABS(M=1)-1
130 LOC.0,8:P."YOUR":G$=INK.(0)
140 IF G$="R" OR G$="S" OR G$="T" T.170
150 IF ASC(G$)=27 T.260
160 G.130
170 X=0:GOS.290:P.MI.(P$,X,9)
180 S=ASC(G$)-81:N(R,S)=N(R,S)+1
190 LOC.15,8:P."MY":G$=CH.(M+81)
200 X=0:GOS.290:P.MI.(P$,X,9)
230 IF S=M T.LOC.10,15:P."A DRAW":GOS.360:G.90
240 W=S-M+ABS(M>S)*3:T(W)=T(W)+1
250 LOC.10,15:P.N$(W):"WIN":GOS.360:G.70
260 CLS:P.:IF T(1)>=T(2) T.P."You loose!":G.280
270 P."You are very clever!"
```



```

280 LOC. 0,20;E.
290 F.I=1 TO 24 ST. 9;IF G$=MI. (P$,I,1) T.X=I
300 N.;RE.
320 P$="ROCK      SCISSORS TISSUE      "
330 DIM T(2),N(3,3),N$(2)
340 R=RND(3)+1;S=RND(3)+1
350 N(R,S)=1;N(S,R)=1;N$(1)="I": N$(2)="
      You";RE
360 LOC. 0,22;P. "Press space bar to continue"
370 A$=INK. (0);IF A$<>" " T. 370
380 LOC. 0,22;P. "
390 RE.

```

程序说明:

1. 10~70 行打印游戏标题和游戏说明。并随机改变背景的色彩,以使显示活泼。
2. 75~80 行打印游戏者和电脑的得分。
3. 90 行的作用是清除上次打印的游戏者、电脑所选内容及胜负结果。
4. 100~120 计算并确定电脑本次所选的内容。它是根据游戏者以前选择的规律算出的。这三行是本程序的关键,它决定了电脑的对策。
5. 130~170 行记录游戏者所选的内容,并打印在屏幕上。150 行为结束转移程序。
6. 180 行记录游戏者本次所选的内容。
7. 190~200 行打印电脑所选的内容。
8. 230~250 行判定胜负,累计得分。其中 230 行是对两者所选内容相同时的处理。240 行对得胜的一方加一分。250 行打印哪一方胜。
9. 260~280 行为结束处理程序。若电脑积分多,则打印“你输了?”;若游戏者得分多,则打印“你真聪明!”

10. 290~300 行是确定双方所选内容名字的全称的子程序。因在选择输入中,只输入石头、剪子、纸的字头 R、S、T,则由该子程序翻译出三种选择项的全称,打印出来。

11. 320~350 行是初始赋值子程序,也可以把它放入主程序的 60 行之后。

12. 360~390 行是场次转换子程序。每进行一场比赛,按空格键再进入下一场。

运行程序,清屏,显示

* GAMBLE GAME *

GUESS ROCK,SCISSORS,TISSUE

BY PRESSING (R),(S),(T)

TO END GAME PRESS (ESC)

1992. 3.

Press space bar to continue

标题内容是“赌博游戏。猜石头、剪刀、纸。依次按 R、S、T 键。结束游戏按 (ESC) 键。按空格键游戏开始”。

按空格键后,清屏,显示

YOUR TOTAL 0 MY TOTAL 0

YOUR ? MY

上一行显示:左边是游戏者(你的)得分,右边是电脑(我的)的得分。

第二行显示你选什么,后跟一光标,等待输入。比如,我们选石头,则键入“R”。这时,电脑若选剪刀,则打印

YOUR ROCK MY SCISSORS

YOU WIN

Press space bar to continue

意思是“你选的是石头,我选的是剪刀,你赢了。按空格键继续。”若按空格键,这时在“YOUR TOTAL 0”一栏中变为“YOUR TOTAL 1”。若第二次我们选石头“R”,电脑也选“R”,则打印“A DRAW”。意思是“不分胜负”。

若欲结束游戏,则按 (ESC) 键,电脑将根据你得分的多少,打印评语。

通过阅读程序和游戏实践可知,电脑能够记住每次游戏者的选择,并把它与游戏者前一次的选择联系起来,进行分析判断,做出击败游戏者的选择。电脑是怎样进行这一判断的呢?我们先看一张表

紧跟着的选择

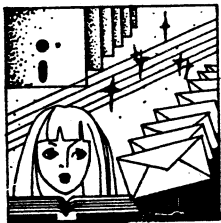
前 一 次 选 择	紧跟着的选择		
	石 头	剪 子	纸
	石 头	0	8
	剪 刀	9	1
	纸	0	3
		3	9

游戏者选择记录表

上表是一个 3×3 阵列的目标表。表中的每一个单元对应了程序中 N(3,3) 数组的一个下标变量。电脑把游戏者每次选择记录在这张表中。轮到电脑选择时,它首先查看游戏者上次选的是什么,习惯于哪种规律的选择,从而选出取胜游戏者的玩法。

假定游戏者习惯于出石头后接着出剪刀,在上表的第一行中记录了游戏者的这种选择规律。即选过石头后再选石头为 0 次;选过石头后选剪刀为 8 次;选过石头后再选纸为 1 次。那么电脑知道游戏者在本次选择了石头,下一次大多数选剪刀,于是它自己就选石头以取胜(下面两行类推)。游戏中,电脑一直盯着这张表,以做出取胜的选择。当然,在实际游戏中,一旦游戏者意识到电脑已经掌握了他的规律,就会选用另外的策略。而当游戏者采用的策略多于一个时,电脑又会相应地选择对付游戏者的策略。不难想象,如果游戏者固定采用一个惯用策略,那么得胜一方将总是电脑。当然,绝大多数游戏者会意识到这一点,及时改变策略。

这一游戏结构比较简单,读者可自行分析,下一讲介绍游戏的程序结构和程序框图。



电子工业出版社软件部最新出版软件

新书与软件

1. XF—通用管理系统 V1.0

本系统有三大特点:具有数据库的主要功能,但不需 CP/M 系统支持及 Z80 卡;响应速度快,全汉字操作;具有齐全、方便的记录和文件管理及统计打印功能。

本软件分为:1. 记录操作,包括查询、修改、删除、恢复、查库和输入共 6 个模块;2. 库操作,包括增字段、删字段、改字段顺序、改字段名、改字段类型、改字段长和库整理共 7 个模块;3. 统计打印,包括合计、统计、查询打印、顺序打印和查库共 5 个模块。此外还有磁盘及文件的基本操作。本软件配有详细的操作说明,可使用户操作一日通。

出版:1992 年 9 月 磁盘:1 张 定价:160 元

2. 工资管理系统

本软件适用于中、小企事业单位进行工资管理,它具有 6 大特点:系统设计合理,功能全面实用,可管理 1400 人的工资档案,针对每人,可使用 24 个数据项和 3 个统计项;运行、存取速度快,数据修改方便;用户界面好,操作简单,提示明确;适用多种打印机型并可进行局部打印;系统还设有保密口令,以使数据资料安全可靠。

出版:1992 年 9 月 磁盘:1 张 定价:160 元

3. CEC—I 英语单词记忆系统

本软件是新一代英语单词学习系统,它有 8 大特点:1. 单词词组随机出现;2. 单词库容量大,可存储 4050 单词;3. 自动时钟计时,它可计算学习时间和效率,使学习成绩量化;4. 有 8 种记忆方式,以适应不同的学习者;5. 8 种记忆方式可单选也可设置自动顺序学习,如设置 1、3、6、7 方式连续学习;6. 可查阅全部学习成绩值:学习时间,测验成绩,学习效率等;7. 可自行设定记忆牢固度值,在学习中系统根据测验的成绩和此值判定单词是否已掌握,对已掌握的单词系统不再提供记忆学习;8. 用户界面友好,操作简单可靠。

出版:1992 年 9 月 磁盘:1 张 定价 48 元

4. 猜谜大奖赛

这是一个集益智、趣味和知识于一体的游戏软件,它配备了适合于儿童、少年、青年和成人的谜语一千余条。这些谜语构思精巧,妙趣横生。对猜中谜语者,软件还提供诗词、格言、音乐、对联、趣闻、笑话及游泳等作为奖品供猜谜者欣赏和娱乐。

本软件设计新颖,谜语全部采用美术字显示,并设有记时、记分和谜底显示功能。此外软件还可进一步增加新谜语。

出版:1992 年 9 月 磁盘:1 张 定价 30 元

(以上 4 种软件仅适用中华学习机)

C-DBAG 中文数据库应用生成器系列新产品

F/D * AG V5.0

北京航空航天大学计算机系(100086) 李昭原

C-DBAG 中文数据库应用生成器是为数据库管理系统开发和使用而研制,其中 2.0 版曾获北京市科技新产品三等奖,4.0 版在全国首先推出网络多用户网络版,F/D * V5.0 是在这些研究基础上,推出的 C—DBAG 系列的最新产品。

自动生成网络多用户 dBASE III plus 或 FoxBASE plus 源程序

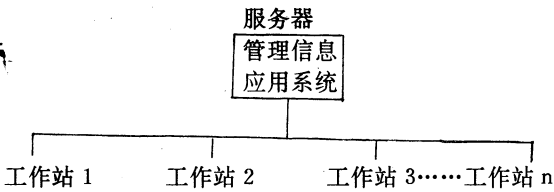
数据库应用生成器,用户在使用时只需按生成器中

中文提示,不需编程,即可生成(如财会、金融、统计、销售等)各种信息管理系统,并且在功能变动时,可任意修改,数据库应用生成器的使用,可大大提高编程人员的工作效率,减轻工作强度,并提高应用系统的可维护性。

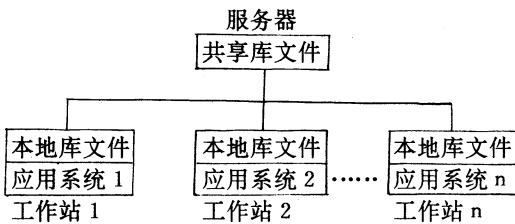
自 4.0 版以后,C—DBAG 产品根据市场需求特别强调,网络多用户应用系统的生成器,并根据不同环境为用户提供两种不同使用方式。

在用户有功能较强主机,管理信息联系紧密的应

用环境下,用户可生成一个应用系统,把其安置在服务器上各个工作站共享这个系统。



而在信息组成复杂,相互独立性较强的应用环境,用户可只将共享库文件安置于服务器中各个工作站可根据自己需要,生成管理共享库,与本地信息库的应用系统。



根据目前用户的需求,F/D * AG5.0 共有以下 8 个功能生成器,用这些生成器即可生成单用户又可生成多用户系统。

下拉式/弹出式菜单生成器

菜单生成器可自动生成工作站独立的或多子系统的系统菜单,当生成至每菜单树叶时,则自动要求生成共享库文件名或本地库文件名菜单,生成菜单理论值为 6 层,各级子系统个数最长达 20 个,菜单屏幕格式及画面生成可自编或系统定义。

数据库文件生成器

库文件结构生成器将建立共享或本地各数据库文件的数据结构,同时可建立代码数据库文件。

数据录入与维护模块生成器

生成对单库和多库进行添加、插入、修改、删除、查询、打印一体化操作程序,用户可选择录入屏幕格式,生成数据快速批量更新操作,生成浏览数据操作,生成从其它库添加操作生成代码库维护操作,生成修改器全操作,生成数据备份操作。

查询模块生成器

生成索引快速查询,匹配模糊查询,组合固定条件查询,组合任意条件查询,任意字段查询结果即可在屏

幕显示也可在打印机上打印。

统计与计算模块生成器

生成分类快速汇总,条件字段统计与计算,分类条件统计与计算,生成条件横向纵向统计与计算生成累计统计,上述各种统计与计算均可在单库或多库上进行。

统计图形生成器

用户可选择曲线图,直方图或圆饼图显示统计结果数据,可大屏幕地纵、横向显示,彩色图形的统计结果数据,并自动标记汉字坐标和统计结果值,统计图形打印程序支持图形打印的放大功能(放大倍数分别为 1、2、3、4、6、8 倍)。

任意格式报表生成器

可生成带横纵向统计与计算,带表头、表体、表层的任意格式报表,用户对字体、字号、报表格式、表头、表体、标题、尾注,左右角标均可自行定义,且可选择和控制打印字段,字段打印顺序,可实现分页打印,非零打印,折行打印,最大宽度可达 300 列。

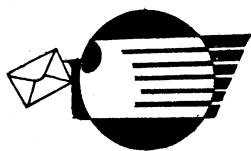
数据字典与文档生成器

由字典生成、文档生成、管理维护三个子系统集成,字典生成器通过扫描一个具体 MIS,可自动生成其库结构字典及程序结构字典。

特点:

1. F/D * AG V5.0 各生成器可单独也可集成使用。
2. 生成器全部由 C 语言编程,具有很强的在线帮助及容错性能。
3. 软件生成全部采用对话方式,易学,易懂。
4. 可生成单用户系统,亦可生成网络环境下多用户信息管理系统。
5. 生成系统全部为 dBASE III plas 或 Fox BASE plas 源程序,用户可修改。
6. 支持“快速原型——增量”开发”方法:便于开发及修改。

C—DBAG 系列产品在现有生成 FoxBASE plas、dBASE III plas 信息管理系统的基础上,将进一步开发其它数据库管理生成器,将要开发的有 ORACLE、SYBASE 等大型数据库管理系统的生成器、将支持 WINDOWS 界面,支持与高级语言的连接,从而大大提高信息管理系统水平,增大应用范围,为大中型厂矿企业的信息管理系统开发提供快速工具。



普及型 PC 个人用户软件 交流联谊活动问题解答(九)

北京中国农科院计算中心(100081) 王路敬

30. 对硬盘进行格式化的目的是什么?

磁盘格式化的第一个目的是根据建立 DOS 分区 FDISK 记录在硬盘上的信息,对属于 PC-DOS 分区的部分进行检查,不再对整个硬盘进行物理格式化;第二个目的是初始化 PC-DOS 的文件分配表和文件根目录区,初始化磁盘结构参数表,如果需要从 PC-DOS 分区引导,则还要建立 PC-DOS 分区引导程序并记录在 PC-DOS 分区的第一扇区,同时建立 IBMBIO.COM, IBMDOS.COM(对长城系列机为 GWBIO.COM, GWDOS.COM, COMMAND.COM 三个文件。

格式化的工作是通过随机提供的 FORMAT.COM 文件来完成的,在操作系统提示符下打入:FORMAT C:/S/V<回车>再根据提示信息进行,用户将看到信息:

```
Format complete
system transferred
```

这个信息告诉用户硬盘格式化操作已经完成并已装入了 DOS 的副本,然后将出下列信息:

```
Volume label(11 characters)
```

用户可打入不超过 11 个字符的卷标名,以后执行 DIR 命令和 CHKDSK 命令时就会显示这个卷标名。如用户不希望给他的硬盘赋一个卷标名,则可直接按回车键。

31. 在 PC-DOS 不同版本中硬盘分区和格式化程序文件的长度有什么不同?

硬盘分区命令和格式化命令的功能随 PC-DOS 版本号增高而增强。一般在 PC-DOS 低版本下分区和格式化的硬盘,在高版本下可以对硬盘进行读写操作,但是在 PC-DOS 高版本下分区和格式化的硬盘,在低版本系统下不能对硬盘进行读写。硬盘分区命令 FDISK 和格式化命令 FORMAT 在 PC-DOS 不同版本中其文件长度有较明显的不同,如表所示:

表 PC-DOS 不同版本中 FDISK 和 FORMAT 文件长度比较

DOS 版本号	2.0	2.1	3.0	3.1	3.2	3.3
命令名称						
FDISK	6177	6369	8076	8173	8173	48216
FORMAT	6016	6912	9015	9398	11135	11616

32. PC-DOS 2.0/2.1 与 PC-DOS 3.0 以后版本的文件分配表有何不同? 对读写硬盘操作有什么关系?

DOS 通过文件分配表 FAT 来管理磁盘空间的分配,反映磁盘空间当前的使用情况。当文件需要空间时,系统将动态地分配一个簇。实际上,FAT 对每个文

件来说起着连接作用。它将磁盘文件实体所占的数据扇区连成一个文件整体。当对硬盘进行读写操作时,硬盘上最小可寻址的单位是一个扇区。通过 DOS 的文件系统功能来读写盘上的一个文件时,DOS 是以簇为单位进行的,簇是磁盘空间的分配单位,一个簇总是对应一个或多个逻辑扇区。盘上的每一簇,FAT 中必须有一项来登记它。PC-DOS 2.0/2.1 中 FAT 表中的每一表项固定为 12 位(即 1.5 字节)长。PC-DOS 3.0 以后的版本里,FAT 的每一表项可以是 12 位,也可以是 16 位。12 位长的表项最多可以表达 4096 个簇,而 16 位长的表项最多可表达 65518 个簇。

当读硬盘上的文件时,FAT 与文件目录一起确定文件在硬盘上的位置;在写文件时,FAT 帮助确定盘上哪些簇是空的,可供分配的。对于一个 20MB 硬盘(GW0520CH 机),磁盘总扇区数 41751 个,采用 12 位长的表项,每簇为 16 扇区。同样是该硬盘,若采用 16 位长的表项,则每个簇分配不到一扇区,即小到 512 个字节。

如果每簇的字节数太小,文件在磁盘上分布过碎,读写文件的速度就会减慢,不宜采取。每簇所包含的扇区数与磁盘的容量有关,而在 PC-DOS 下每簇扇区数是由 FDISK 和 FORMAT 命令确定的,且以后不能改变,除非再一次运行 FDISK 和 FORMAT 命令。在 PC-DOS 3.0 以后版本里 FDISK 命令建立 DOS 分区时,究竟是使用 12 位还是 16 位表项来表示簇号,取决于 DOS 分区是大于还是等于 20740 簇,大于或等于则所建立的 FAT 的每一表项长为 16 位,否则为 12 位。

33. 新一代长城机基本配置硬盘的型号、主要参数及 CMOS 设置类型是什么?

掌握新一代长城机 GW0520DH、GW286B、G286BH、GW286EX、GW386/20 基本配置硬盘的型号,主要参数及 CMOS 设置类型是正确设置系统配置的一项重要内容,必须了解。如下表所示:

机 器 型 号	硬 盘 型 号	设置容量 类型(兆)	硬 盘 参 数				
			最大 磁道数 CYLS	磁头数 HEADS	每道 扇区数 SECTORS	予补 偿值 PRECOMP	
GW0520DH	HH—825	—	20	615	4	17	300
GW286B	HH—830	3	30	615	6	17	300
	ST—238R	3	30	615	6	17	300
GW286BH	HH—1050	47	40	1024	5	17	NONE
GW286EX							
GW386/20	ST251—1	40	40	820	6	17	NONE

如是 GW286BH、GW286EX、GW386/20 等机器,当配置 ST251—1 时,其机器出厂流水号尾部有英文字母“S”,若无“S”的机器,如无特殊说明,则配 HH—1050 硬盘;所有配此硬盘的机器,如其 CMOS 硬盘类型 47 的各参数与表中参数不符,请选用类型 11。

34. 在什么情况下,从硬盘引导操作系统时,进入 ROM BASIC 解释程序?

当从硬盘上启动操作系统 PC—DOS 或 CC—DOS 时,有时启动失败,进入 ROM BASIC 解释程序,出现这种情况通常有下列原因:

(1)当系统既没有硬盘又没有在软盘驱动器 A 中插入系统盘时,将使启动失败,而进入 ROM BASIC。因为 DOS 启动时,首先检查 A 驱动器有无 DOS 盘,若有,则从 A 驱动器将 DOS 装入内存,若无 DOS 盘,则判断 C 盘上有无 DOS,若有,则从 C 盘装入内存,A 驱动器没有 DOS 盘,C 盘也没有 DOS,则进入 ROM BASIC 解释程序。

(2)既不能正确读出软盘第 1 扇,又不能正确读出硬盘第 1 扇。因为软盘的 0 面 0 道 1 扇区是引导扇区,在系统启动时用它来查找和装入 PC—DOS 的两个隐含文件 IBMBIO.COM 和 IBMDOS.COM,并把控制权交给 IBMBIO.COM,它的作用是首先检查驱动器 A 上是否插入了系统盘,若是,则引导 DOS 进入内存,否则给出出错信息并进入 ROM BASIC 解释程序。除此之外,还要检查上述两个隐含文件的顺序对不对,正确的顺序是 IBMBIO.COM 在先,IBMDOS.COM 在后。如果这个顺序不对也要给出出错信息。硬盘的第 1 扇区是主引导程序和分区信息表。硬盘分区表描述了硬盘分区状况,主引导程序供硬盘自举使用。该扇区的前 240 个字节是主引导记录,从 1BEH 位移开始的 64 个字节是硬盘分区表,最后两个字节是 AA55H,是自举记录的有效标志字。

(3)没有在驱动器 A 插入 DOS 盘片或不能读出软盘的第 1 扇区,而硬盘上第 1 扇区最后一个字不是 AA55H,也进入 ROM BASIC 解释程序。

(4)没有在驱动器 A 插入 DOS 盘片或不能读出软盘的第 1 扇区,而硬盘上又没有可引导的分区,也会进入 ROM BASIC 解释程序。

35. 在何情况下,从硬盘启动系统失败而进入死循环?

在下列条件之一发生时,系统引导失败,进入死循环:

(1)没有在软盘驱动器 A 插入 DOS 盘片或不能读出软盘第 1 扇区,而硬盘有多个引导分区。一个硬盘系统可分为 4 个“分区”,划分这些分区的目的是支持 DOS 以外的操作系统。因而可在同一个硬盘上驻留 DOS 以及其他磁盘操作系统,并且都可工作,最多可容纳四种操作系统,每种操作系统在硬盘上建立自己专用的分区操作时都有一个相应的实用程序。分区的具

体数目和 DOS 分区的大小可由 DOS 外部命令 FDISK 设置。每一个分区都有一个对应的分区表,由 16 个字节组成,若第 0 字节值为 80H 时表示该分区可自举,其值为 00H 时,表示该分区不可自举。当硬盘上引导标志多于 1 个时即显示:“Invalid Partition Table”,进入死循环。

(2)没有在驱动器 A 插入 DOS 盘片或不能读出软盘第 1 扇区,而硬盘 PC—DOS 分区引导程序又读不出来。通常在 DOS 分区的首扇区上存放 DOS 引导记录,当硬盘主引导记录搜索到 DOS 引导记录扇区时,便将控制权交给 DOS 引导记录,由它负责将 DOS 装入内存,这一过程和软盘启动相同。当硬盘 PC—DOS 分区引导程序读不出来时,当然使启动系统失败,此时会显示:“Error Loading Operating System”,使系统进入死循环。

(3)没有在驱动器 A 插入盘片或不能读出软盘第 1 扇区,而硬盘 PC—DOS 分区引导程序的最后一个字不是 AA55H。一个硬盘 PC—DOS 分区引导程序正常的情况下,其最后一个字的目标码应当是 AA55H。若不是,则显示“Missing Operating System”。

(4)不能把 IBMBIO.COM 文件读入内存。IBMBIO.COM 是 ROM 中 BIOS 的低级接口模块,在系统启动过程中,IBMBIO.COM 负责测定系统中设备的状态,初始化附加设备,使磁盘系统复位,设置低序号的中断向量,解释 CONFIG.SYS 文件并设置系统环境,加载可安装的设备驱动程序以及给引入内存的 IBMDOS.COM 重新定位。不能把 IBMBIO.COM 文件装入内存,将显示:“Disk Boot Failure”,从而进入死循环。

36. 开机后没有显示“1701”号错,但每次企图转 C 盘时屏幕只是显示:“无效驱动器参数”,是什么原因造成的?

出现这种情况一般有两种可能的原因造成。其一是硬盘控制器上的 ROM BIOS 程序被禁止了或有其他错误。硬盘 ROM BIOS 主要提供硬盘输入输出驱动程序,它的组成包括以下几个组成部分:

(1)硬盘输入输出参数的定义,其中包括:错误代码参数,设备地址端口,硬盘控制器命令字。

(2)硬盘数据区。它包括在 0 段设置磁盘中断向量号,在 40 段设置磁盘状态字。

(3)硬盘输入输出的初始化程序。

(4)自举装入程序——INT 19H。

(5)硬盘输入输出驱动程序 INT 13H。

硬盘通过硬盘控制器的 ROM BIOS 进入系统的 ROM BIOS,从系统的 ROM 中取得对硬盘的输入输出请求的控制。

第二个原因:是硬盘上第 1 扇区的 DOS 分区信息或有关磁盘参数受到破坏。引起这种故障常常是由于主机电源容量不足,使得在长期工作时,主机突然掉电自动保护,接着又立即加电开机,这一断一通,对于正在读写的控制器和驱动器容易造成某些信息的丢失或

误写。为此作为主机的日常性维护,其中有一条要求开、关机之间时间间隔不要太短,即频繁的开关机是造成硬盘损坏或信息丢失一个重要原因。

37. 硬盘使用一段时间后读写文件时容易发生错误,是什么原因?

从使用的实践经验上来看,其原因既有介质方面的,也有磁头定位精度方面的,但使用方面的原因是一般用户尤其要注意的。对于 40MB 以下的大多数硬盘,在断电之后不会自动退回到非数据区磁道,关机之前磁头停在哪里,关机之后仍停在哪里,并和盘面接触。如果主机在这时搬运受到冲击,硬盘的磁头有可能擦伤硬盘盘面而影响读写。一旦出现这种情况需作格式

化的操作,才能重新建立硬盘系统。

38. PC-DOS 对硬盘容量是如何限制的?

由硬盘 PC-DOS 分区引导程序中的硬盘基本输入输出参数表分析,该表由 19 个字节组成,其中有二个字节表示硬盘的总扇区。二个字节为 16 个二进制数,可以表示最大数为 65536,因此,一个硬盘最大不能超过 65536 个扇区,大约 33MB。如果一个硬盘的容量大于 33MB,则需要把它划分为几个逻辑盘,而每个逻辑盘不能超过 33MB 最大限数。例如长城 286BH 配置硬盘为 40MB,可划分 C 盘和 D 盘两个逻辑盘,C 盘最大可为 33MB。

网络管理信息系统快速开发工具 支持“快速原型——增量”开发方法

C-DBAG 中文数据库系列应用生成器新产品

F/D * AGV5.0

北京航空航天大学开发

使用本系统只需按照提示输入用户需求,不需编程,即可生成(如财会、金融、统计、销售等)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

详细功能请看本期软件介绍栏目

组成: • 下拉式/弹出式菜单生成器

- 数据库文件生成器
- 数据录入与维护模式生成器
- 查询模块生成器
- 统计与计算模块生成器
- 统计图形生成器

• 任意格式报表生成器

• 数据字典与文档生成器

• 92 年 10 月 27 日在北航开办培训班

经销: 中电华北公司电脑部 电话:81.1810

地址:北京万寿路西街五号

邮编:100036

联系人:石立军 魏 国

北京景文科技开发公司

电话 832.2255 转 458

地址:北京西直门外首体主楼 305

邮编:100081

联系人:袁凯峰 李应知

本公司经营计算机及外设通讯设备,欢迎来电来函联系。