

E & C

1993

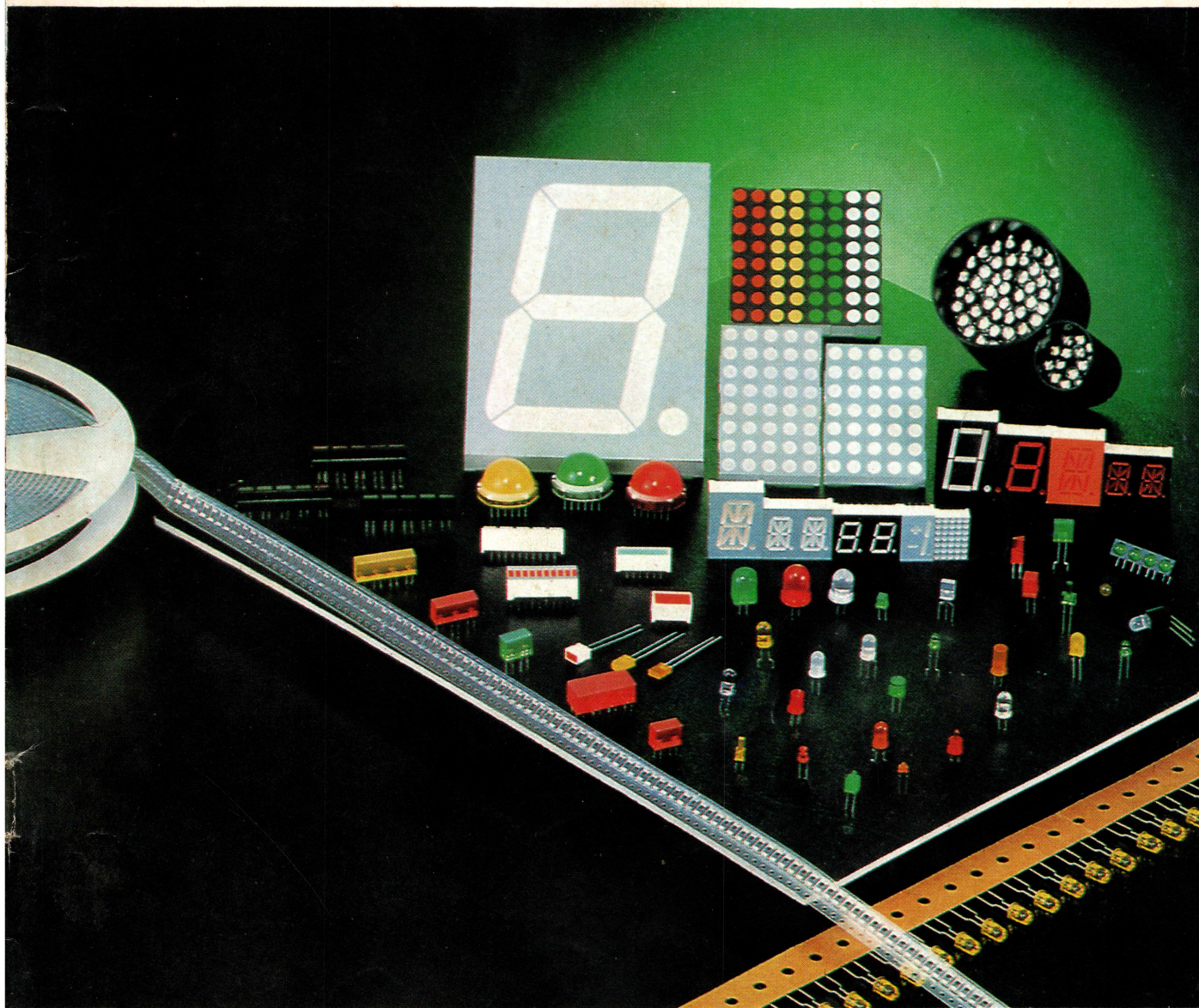
●一九九三年 ●总期第98期

5

電子

ISSN 1000-1077

與 電腦



• ELECTRONICS AND COMPUTERS •

NEPCON SHANGHAI '93

5TH INTERNEPCON/SEMICONDUCTOR SHANGHAI '93
第五屆上海國際電子生產暨半導體工業展覽

25-28TH MAY 1993, SHANGHAI EXHIBITION CENTRE,
SHANGHAI, PEOPLE'S REPUBLIC OF CHINA.



从三种通用的八位单片机硬、软件比较来看 Z8 系列单片机的优越性

表 2 七程序运行比较结果

测试内容	MC6801(4MHz)		8051(12MHz)		Z8(8MHz)		Z8(12MHz)		相对比较	
	时钟	时间	时钟	时间	时钟	时间	时钟	时间	MC6801	8051
CRC	367	367	139	139	546	137	546	91	4.03	1.53
搜索字符	687	687	280	280	1524	382	1524	254	2.70	1.10
计算 GoTo	110	110	75	75	228	57	228	38	2.89	1.97
右移 5 位	61	61	46	46	154	38	154	26	2.35	1.78
移 64B 数据	2306	2306	577	577	1924	481	1924	321	7.18	1.80
调用子程序	14	14	4	4	34	8.5	34	5.7	2.46	0.70
端口置数	8	8	2	2	10	2.5	10	1.7	4.71	1.18
综合性能比较									3.76	1.44

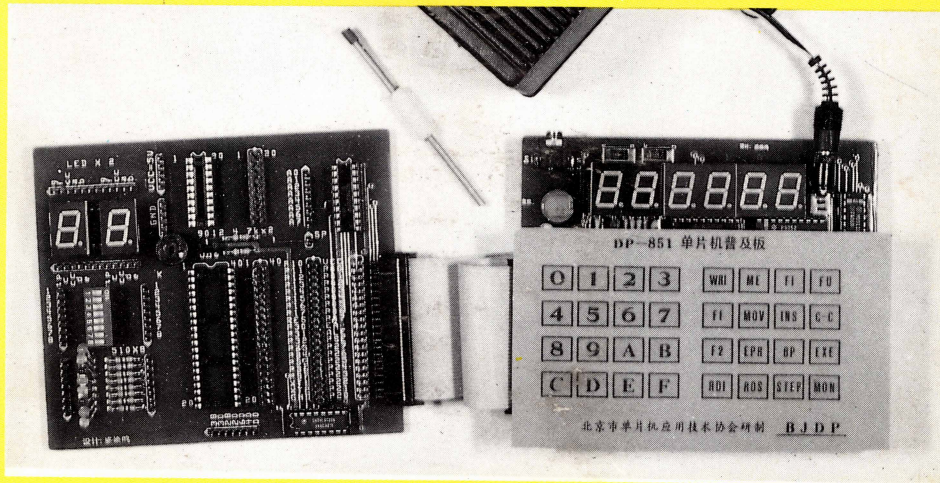
表 3 字节,指令,时间比较

	字节			指令			时间(μ s)		
	MC6801	8051	28611	MC6801	8051	28611	MC6801	8051	28611
CRC	28	31	22	15	18	12	367	139	91
搜索字符	15	15	11	8	7	5	687	280	254
右移 5 位	11	15	9	6	9	5	61	46	26
计算 GoTo	17	21	15	8	12	7	110	75	38
移 64B 数据	21	10	6	11	7	3	2306	577	321
端口置数	6	3	2	3	1	1	8	2	1.7
调用子程序	3	4	3	3	2	2	14	4	5.7

(董伯明提供)

单片 851,“八五”科技潮

开展函授教学,易学、实用、比赛 请您选择单片机普及板 DP—851



一. DP—851 单片机普及板的外型结构(专利号:93301539.9)

二. DP—851 单片机普及板的功能特点

- ①**开发系统板**:主板+键盘板(或主板+pc机)可构成单板型单片机开发器。主板上具有16KB的监控程序(键盘和PC机各占8KB),具有置入、修改、调试、运行等多种开发功能。
- ②**教学实验板**:开发器+实验板可构成教学实验板。该板从教学的实际需要出发,具备了实验内容多、覆盖范围宽、结构灵活、性能可靠、价格低廉等优点,是一种大众型的教学实验设备。
- ③**目标应用板**:开发器+A/D,D/A等扩展板可构成用户的目标系统。

三. DP—851 单片机普及板的培训计划

- ①函授:由中国计算机学会和北京市单片机应用技术协会面向全国开展函授教学。
- ②比赛:计算机学会于九四年底用DP—851单片机普及板进行全国学装微电脑比赛。

四. DP—851 单片机普及板的销售价格:398元/套

- ①398元包括:薄膜键盘板;主板;实验板;说明书;电源;工具;包装盒;40芯电缆。
- ②可选件价格:pc联机电缆和盘片:50元/套;自升压EPROM固化板:190元/块;三位半A/D板:120元/块;四位半A/D板:240元/块;积分12位A/D板:350元/块;高速574A/D板:980元/块;A/D、D/A板:320元/块

开发单位:北京市单片机应用技术协会(BJDP)

经营单位:北京市技术交流站 开户行:工商北京珠市口分理处(帐号:089632—30)

地址:北京市虎坊桥13号(邮编:100052)联系人:苗淑萍(电话:3035931—2085)

电子与电脑

一九九三年 总期第 98 期

目 录

· 综述 ·

商业管理软件:〈流水账〉的写作经验 蔡培强(2)
对公用数据网的几点建议 薛兴华(3)

· PC 用户 ·

西文图形状态下显示汉字的 Turboc 程序
..... 张辉 张敬民 范俭(4)
PC Tools 若干高级实用技巧 李日林(6)
DOS 下图象文件的打印输出 杨晓慧(7)
如何使 PC 机具有实时时钟 李维宪(9)
再谈运行 FoxBASE+ 内存不足的问题 冯建军(10)
利用电脑预测儿童青少年成人时的身高
..... 黎彦才 米绍平 徐奎(11)
五笔字型系统用压缩字库 徐庆恩(13)
TRAVELLER(C) 病毒的发现与清除 桂宁(15)
CPAV——使您的电脑百毒不侵 郭文典(16)
用“锁匙软盘”保护硬盘 崔来堂(17)

· 学习机之友 ·

启动 DOS 的技巧 邵爱柏(18)
中华学习机 DOS 使用技巧 张本灵(19)
学习机游戏软件“攻关秘诀” 王海涛(20)
任意数的高精度除法运算程序 张新莲(20)
ProDOS 系统内部结构剖析(续) 廖凯(22)

· FORTH 语言讲座 ·

第 6 讲 程序控制结构(上) 丁志伟(24)

· 初、中级程序员软件水平考试辅导 ·

C 语言试题与分析 赵国瑞(28)

· 学用单片机 ·

BJS—51 教学实验板的普及型新品“DP—851 单片机普及板”
技术讲座 王玉民(31)
单片机普及函授班招生简章 (32)
SCB 系列单片单板机介绍 (32)
从三种通用的八位单片机硬、软件比较来看
Z8 系列单片机的优越性(续) 董伯明(33)

· 电脑巧开发 ·

与 CEC—I 微机接口的 EPROM 仿真卡 王正英(34)
PC 机并行口的扩展 孙胜春 邢昌凤 肖英(37)

· 维修经验谈 ·

彩显工作原理简述和故障诊断(下) 胡野红(38)

· 电脑游戏机 ·

第六讲 游戏机程序实例分析(上) 于春(41)
家用电子游戏机故障维修一例 杨林(43)

· IC 电路应用 ·

RS232/TTL 电平转换芯片 MAX232 的原理及应用
..... 罗维国 吴微(46)
单片功率开关电源 IC 李秀华 吴剑裳(48)

· 电脑通信 ·

应用程序接口(API)时代的开端 文海(50)

· 传真机专题讲座 ·

传真机原理(下)——调制和解调
..... 张建军 孙晓春 张景生(51)

· 读者联谊 ·

双频卡使用问答 程伟(55)
两处疏漏 一点建议 何崇乐(56)

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 特约编审:苏子栋

责任编辑:杨逢仪

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:1.60 元

商业管理软件：〈流水帐〉的写作经验

新加坡 FOX:065—4525802 蔡培强

俗语说：“一手交钱，一手交货”。在真实的商业社会里，这样的情形是很少的。往往一宗交易从开单的那一刻起，要经历许多如：换货、退货、扣钱、拖账、对敲、支票不兑现等等繁琐的手续后，才算结束。这也是为什么理账的小姐薪水总是比只会打字的小姐多，因为理账必须有专门的知识。

自从电脑掉到和电视机一样的价钱后，几乎人人都想买电脑了。虽然大多数电脑最后都成了高级游戏机，但也有部分的商家是因为想认真把业务电脑化才买电脑的。他们有些买了一些外国的现成软件来用，有些还请人用数据库管理系统专门替自己公司写一套专用软件。这些软件的确也发挥了一定的作用，但他们都有某些局限。外国软件总有一些不适合本地使用的东西，尤其很多华人家庭生意，都有他们自己一套生意手法，如果样样都要循规蹈矩，肯定无法办到。请人专门编写的软件（所谓定制软件），花费要比现成软件贵很多很多，而且不一定完善。因为每个程序员每个月要多少收入是有公认的数目的，你能付多少，他就为你做多少；你要的他会替你做，你没提到的他不会多做。等你用了以后发现少了什么又要请他回来。从程序员的角度看这不是什么好差事，叫做手尾多多，没完没了，穷其一生精力也写不了几个完善的软件。

就这样，市场出现了许多土生软件。这些软件比较能配合本国的现实需要，开发期也比较长，通常一两年时间，还要经过用户试用，消除所有错误后，才广泛销售。为了适应象新加坡许多用户的一般需要，也就不一定能完全适合个别用户的特别需求。软件的“软”度关系到有多少用户会接受它，而能够做到多软却取决于编程员的功力、经验和想象力了。笔者曾经编写过这样的软件，前后花费了两年多的时间，愿将这方面的一点心得与各位分享，期望能有抛砖引玉之效。

首先是语文问题。长期以来人们总以为电脑只懂英文，那是误会。电脑什么语文也不懂，它只会将你写给它的文字照样显示出来。为了适应象新加坡这个多元语文的社会，软件必须能分别载入英文、马来文、简体中文、繁体中文甚至印度文，并且能在软件运行中变换。比如一间五金公司，做账小姐要用英文入账，老板要用中文查账，软件必须能让他们自由选择。这样做并不太难。你只需将程序中所有说明的字串收录在一个外部文件中，然后复制成几个文件，加以编号，再翻译成其他语文，就能制成多种语文的软件了。要注意的是，一些程序用来作判别的字串就不能这样做，否则软件在不同语文下运行就会有不同的结果。

第二是松紧问题。很多商家不敢用电脑理账是担

心电脑太过“准”，很多惯用的手法在电脑化后都不能以再用了。比如改单，原本只需在账单上画一画，再交给理账小姐去理就好，有些软件却不允许改单，一律要用开信用单(Credit note)的方法才行。事实上，电脑不断重算是不会累的，所以允许修改资料是可行的，只需设计一段反向运算的程序，就能带给用户很多方便。再比如库存量，有些公司经营了几十年，库存从来没有准过，如果一开始就要求输入正确的库存资料，甚至在库存量出现负数时不准开单卖货，虽然逻辑没错，却不符合实际。所以软件必须允许在不合理情况下继续运行，只将情况作标记，在作报告时显现出来。当然，如果一味地松下去也是不对的，所以经过一段时日，就要逐步收紧，最终达到绝对准确的目标。

第三是调整问题。要使一个软件能同时适合不同的用户，必须在软件的性能调整方面做足功夫，要尽量使用变量，并由外部的控制文件输入数值。这样只需修改控制文件的内容，就能调整软件的性能，以适合用户的需求。控制文件还可以用来开关软件的某些功能，使到对一些用户没有用到的功能不会出现，减少操作程序。

第四是输入问题。有很多的软件在设计上总喜欢用编号去组织资料，无论公司名、货物名都要编号。那是因为号码是一种最容易辨认的排序方式，电脑也最容易处理。但对用户来说就不容易了，要为几千种货物编号，脑力花费不在话下，就算编好了，平时都很熟悉的东西全变成H—101、K—303等陌生的号码，还要花一段时间去背熟。其实以现在电脑的处理能力，是不必采用号码去组织资料的，所以应该让用户以原来的名字去输入资料，电脑花时间去建立索引。如果需要输入中文，还必须能记忆每个曾经输入的名字，做到一次输入，终生受用，因为不是人人都懂得输入中文的。

第五是输出问题。电脑化的目的就是要提高效率，所谓事半功倍也，所以必须尽可能利用输入的资料，通过大量的运算产生相关的资料，让用户使用。比如利用买卖货品的数量、价格，为用户提供采购咨询，减少货仓积压；还可作利润统计，知道什么好赚多做，什么难赚少来。而利用顾客还账时间及欠债数额，可分析其还账能力，减少呆账和坏账。同样的资料还可以查出某样货品卖给某公司是多少钱，卖给另一间公司又是多少钱，避免报价错误。总之要将输入的资料用尽，才能充分显现电脑化所带来的好处。

第六是维护问题。由于软件是要写给很多用户使用的，所以有必要维持一致性。不能因为新用户的特别要求而随便增加程序，这将损害其他用户的利益，不但

多用了内存,也增加出错的机会。也不能随便另写独立的小程序去应付,不然以后版本升级时还要为那些小程序烦恼,得不偿失。所以,遇到坚持要有特别功能的新用户,最好介绍他用定制软件,而跟他说:再见!

就说再见吧!

对公用数据网的几点建议

江苏省邮电管理局(210003) 薛兴华

我国为赶上世界信息技术的先进水平,近几年明显加快了公用数据通信网的建设,采取了“利用用户电报网发展低速数据业务,利用公用电话自动交换网发展中速数据业务,积极开发分组交换技术,建立国家公用数据通信网”的技术政策。目前,我国用户电报及低速数据网已建成,300bps(比特/秒)业务可开通至所有省会城市及大部分地市;公用分组交换数据网(CNPAC)已对外全面开放业务,部分省级数据网正在加紧兴建;新引进的美国休斯公司的小天线地球站(VSAT)卫星数据通信正式投入运营;许多单位利用电话网或租用专线开通了中速数据业务等,全国公用数据网已基本形成。

邮电部从法国 SESA 公司引进的 DPS25 I 型分组交换系统组建的我国第一个公用分组交换数据网(CNPAC),于 1989 年 11 月组网运行,开始向社会开放数据通信业务。首期工程在北京、上海、广州设立了三个节点交换机;在武汉、南京、沈阳、西安、成都、天津、深圳及邮电部数据所设远程集中器;网管中心(NMC)和国际出口入口局均设在北京。系统采用 CCITT 建议的 X. 25、X. 75、X. 28、X. 29、X. 121 等国际标准规程,并可执行 IBM 公司的 SNA/SDL 规程,以便和 IBM 计算机相连。

CNPAC 可以为该 10 个城市的计算机和各类终端之间的通信提供服务。这 10 个城市以外的计算机可以通过长途通信线路电话接入该网络进行通信。

CNPAC 投入运营后,在社会提供数据服务方面发挥了重要作用,业务量不断增加,用户发展很快。但是也存在一些问题,影响了数据网的进一步发展。除了联机数据库资源极少外,还有线路质量差、分组交换网覆盖面小、长时间和大批量的数据交换不经济等问题。进一步发展和完善我国的公用数据网,可从如下几方面着手。

1. 不断扩大 CNPA 覆盖面,将分组交换网延伸至全国各省会城市及部分大中城市。

现在全国有微机 70 多万台,由于系统部门、地区和专业的限制,很难做到部门或行业、地区之间的统一管理和协调一致,加上网络不发达,致使 90 年代先进的计算机数据通信优势没有发挥出来。CNPAC 投入运营后,在社会提供联网数据服务方面发挥了重要

作用,业务量不断增加,用户发展很快。北京、上海、广州等城市开放业务仅两年,同步用户端口已趋于饱和。江苏、安徽等省在去年的抗洪防涝救灾中,已充分认识到数据通信不同于话音通信的重要作用。水利、交通、银行等部门纷纷要求租用专线和利用公用数据网组建本部门的系统网。只有少数几个大城市能进行数据业务的 CNPAC 已限制了我国计算机通信进一步发展,很多用户因 CNPAC 覆盖面小而不能进网,要求通过公用分组交换网来实现本部门的计算机联网十分迫切。为此,邮电部将在去年扩(改)建 CNPAC 系统,年底开通,可满足 1992~2011 年的 20 年的计算机数据通信业务发展需求。

CNPAC 扩(改)建工程由骨干网和地区网组成。骨干网由一个网络管理中心、32 个节点交换机、5000 多个端口构成。网络中心设在北京、上海、南京等 31 个省会城市和重庆市。骨干网采用不完全网状结构。其中 8 个大区汇接中心北京、上海、南京、沈阳、武汉、西安、成都、广州之间采用网状结构。其它交换中心根据业务流量流向预测、网路的安全性、可靠性及传输通路可能提供的优质电路等条件来确定交换中心之间是否设中继电路及所需中继电路的数量、速率等。

地区网。各省、市、自治区根据本地数据业务发展需要,可通过设置在各省、市、自治区骨干网上的交换中心将网络延伸至本地区的地市及县一级,组成地区性公用分组交换数据网,以进一步扩大网络的服务面。率先建成的 JSPAC 就是一例。地区网的建成将改变只能通过电话网进行点对点或电路交换的方式进行数据通信的限制,可满足各行各业对数据通信和数据库资源共享的需要。

2. 加快用于公用数据通信网内的数据库开发,各部门建成的数据库应尽可能经过公用网对用户开放。

实现计算机和数据库资源的充分共享是兴建公用分组交换数据网的重要目的,如果公用数据通信网拥有大量的、丰富的数据库资源,用户就能通过网络实现共享,发挥出现代通信手段的优势。

我国目前 90% 以上是各部门自建自用的数据库,仅有 6% 左右建库的目的是为国内外用户提供服务的,社会联机数据库服务很少,与我国计算机系统建设的 250 亿元投资规模很不相称。国内没有专门组织数据库销售与服务的企业,应尽快按照商品经济机制,大力发展数据库公共服务业,成立相关专业公司,振兴我国用于联机网络的数据库产业。全国数据库建设应统一规划,加强管理,积极交流,避免低水平的重复劳动。

3. 加快开发国产设备,尽快在全网的支线上启用国产设备。国内设备的研制应从小到大、从外围设备到核心,先研制小容量的交换机、集中器、PAD、网络设备及分组终端,再逐步开展中、大容量的交换机和网管中心。

4. 在建设 CNPAC 和省级分组数据交换网的同时,加紧进行计算机网络、数据库的标准化研究与制

(下转 49 页)

西文图形状态下显示汉字的 Turbo C 程序

东北工学院机械二系(110006) 张辉 张敬民 范俭

随着计算机技术的迅速发展,微型机已在国内各行各业得到了广泛的应用。由于大量用户的英语水平有限,因此,软件中的汉字提示是广大软件设计人员针对国内用户要考虑的一个关键问题,要求所开发的软件具有汉字显示进行人机对话的功能。

实现汉字提示的传统方法是首先进入汉字操作系统,软件在汉字操作系统的支持下才能显示汉字。但这一方法的致命弱点是当进入汉字操作系统后,汉字的点阵信息要占据相当大的内存空间(100~230KB左右),致使一些需要较大内存的软件设计不得不采用内外存交换的办法,降低了软件的运行效率和可靠性。

我们在开发 CAD 系统的过程中发现,大部分的软件的汉字处理量并不象汉字处理软件那样频繁和庞大,仅需少量的汉字提示。根据这一特点,本文提出了用 Turbo C 在西文图形状态下直接显示汉字的方法。使用该方法显示汉字完全脱离了汉字操作系统,几乎不占用计算机的内存空间,解决了汉字显示占用大量内存的问题。该法在某一 CAD 软件开发中取得了良好的效果。

一、汉字字模的获取

要在软件中显示汉字,必须首先获得汉字的字模(点阵信息)。单个汉字的显示一般由 16×16 点阵图形组成,汉字字模的排列是每行 16 点,占 2 个字节,共 16 行占 32 字节。一个汉字的满点数为 256 点,每点占 1 位,某位为 1 时有信息,由有信息的点便组成汉字或字符的图形。这些点阵信息按一定顺序存放于一个磁盘文件中,我们可在任一 CCDOS 汉字操作系统中找到,它通常的名字为 CCLIB。

汉字在内存中的代码是 2 个字节组成的机内码。因此,在显示汉字点阵信息之前首先要获得要显示汉字的机内码,然后才能根据该机内码计算出该汉字字模在磁盘文件中的位置,得到点阵信息。获取汉字机内码的最简单方法是在输入源程序时在汉字操作系统支持下输入要显示的汉字,当源程序输入完毕后即可退出汉字操作系统,进入西文 DOS。

汉字的点阵信息在文件中是按区位码顺序存放的,每个区存放 94 个汉字,1~9 区为图形符号区;10~15 区为空白区,没有点阵信息;汉字的起始区为 16 区。因此,根据汉字机内码计算汉字字模在文件中的位置的方法如下:

假设某一汉字的机内码的两个字节为 ch1 和 ch2 (ch1 为第一字节),则:

$$\text{区位号} = (\text{ch1 and } 7\text{FH}) - 20\text{H}$$

$$\text{偏移量} = (\text{ch2 and } 7\text{FH}) - 20\text{H}$$

当要显示的为 1~9 区的图形符号时,该字前的符号总数为:

$$\text{字符总数} = (\text{区位号} - 1) \times 94 + (\text{偏移量} - 1)$$

当要显示的为 16 区以后的汉字时,由于 16 区前有 6 个空区,所以其前的汉字总数为:

$$\text{字符总数} = (\text{区位号} - 1 - 6) \times 94 + (\text{偏移量} - 1)$$

需要注意的是有的 CCLIB 中没有 9 区的制表符号,而将其放在了 6 区,所以汉字前的空区数为 7,故上式中的 6 应换为 7。

16×16 点阵汉字由 32 字节表示,所以要显示的汉字字模的首字节在字库文件中的位置为:

$$\text{字模位置} = \text{字符总数} \times 32$$

确定字模位置后,就可将文件指针移至该位置,读取汉字点阵的 32 个连续字节。

二、汉字的屏幕显示

Borland 公司的 Turbo C 是目前国内外在微机上海运行的最为流行的 C 语言版本,它为软件开发提供了丰富的屏幕操作与图形功能函数。其中的 getimage() 和 putimage() 两个函数为我们在西文图形状态下直接显示汉字提供了极大的方便。

getimage() 函数用于复制图形视口的某个矩形区域到内存缓冲区,putimage() 函数是将内存缓冲区中的内容复制到屏幕上。但是由于我们要显示的汉字还没有出现在屏幕上,所以就不能使用 getimage() 函数填写供 putimage() 函数显示的内存汉字图形数据,为此我们研究了用 getimage() 函数得到的屏幕图形数据格式,找出了按其格式填写汉字信息的方法,然后调用 putimage() 函数,将按此格式的汉字字模送到显示内存映象区快速显示。下面我们就来分析 getimage() 函数得到屏幕数据的格式及汉字点阵信息的填写方法。

getimage() 函数的调用形式为:

getimage(int left, int top, int right, int bottom, *buf)

其中:left—箱形区域的左上角 X 坐标;

top—箱形区域的左上角 y 坐标;

right—箱形区域的右下角 x 坐标;

bottom—箱形区域的右下角 y 坐标;

* buf—指向存储屏幕图形的内存指针。

在内存中指定屏幕区域的图形数据按行存放,一行的每 8 个点占用 1 个字节,如果一行上点的个数不是 8 的整数倍时,最后一个字节的其余位用 0 添入。当

屏幕状态为单色(或黑白)时,每行所占的字节数为:

$$\text{BYTE}=(\text{right}-\text{left}+8)/8$$

当屏幕状态为 16 色时,则需 4 个彩色平面,这时每行所需的字节数为:

$$\text{BYTE}=(\text{right}-\text{left}+8)/8 \times 4$$

则图形数据所需字节总数为:

$$\text{Total BYTE}=\text{BYTE} \times (\text{bottom}-\text{top}+1)$$

所需内存缓冲区字节总数为:

$$\text{memory}=\text{Total BYTE}+6$$

这是因为在缓冲区的第 1 和第 2 字节要存放(right-left)的值;第 3、第 4 字节存放(bottom-top)的值。缓冲区的最后 2 个字节为无意义字节,其余为图形数据字节。

因此,对于单色(或黑白)屏幕状态,缓冲区的数据格式为:

- 1,2 字节—(right-left)值,低 8 位在前;
- 3,4 字节—(bottom-top)值,低 8 位在前;
- 5~Total BYTE+4——图形数据,按行排列;
- 最后 2 个字节——填入 0;
- 当屏幕状态为 16 色时,其格式为:

- 1~4 字节同单色屏幕状态;
- 5~Total BYTE+4—图形数据,按行排列;每行中的数据又分为 4 组,每一组代表一个彩色平面上该行的点阵信息;第 1 组代表彩色平面 I,第 2 组代表 R,第 3 组代表 G,第 4 组代表 B。这 4 个彩色平面组成 16 种颜色;

最后 2 个字节同单色屏幕状态。

了解了 getimage()函数产生的屏幕图形数据格式后,我们就可以按以上格式填写汉字点阵信息了。一个汉字由 16×16 点阵组成,因此,每一个汉字在单色屏幕状态下所需的缓冲区字节数为:

$$\text{memory}=32+6=38(\text{字节})$$

假定字模的 32 个字节为 D₀~D₃₁,则屏幕缓冲区的数据为:

```

0F 00 0F 00 D0 D1 D2 D3 D4 D5
D6 D7 D8 D9 D10 D11 D12 D13 D14 D15
D16 D17 D18 D19 D20 D21 D22 D23 D24 D25
D26 D27 D28 D29 D30 D31 00 00

```

16 色屏幕状态下的缓冲区总数为:

$$\text{memory}=32 \times 4+6=134(\text{字节})$$

若要显示红色汉字时,其数据格式如下:

```

0F 00 0F 00 00 00 D0 D1 00 00 00 00
00 00 D2 D3 00 00 00 00
00 00 D4 D5 00 00 00 00
00 00 D6 D7 00 00 00 00
00 00 D8 D9 00 00 00 00
00 00 D10 D11 00 00 00 00
00 00 D12 D13 00 00 00 00
00 00 D14 D15 00 00 00 00
00 00 D16 D17 00 00 00 00

```

```

00 00 D18 D19 00 00 00 00
00 00 D20 D21 00 00 00 00
00 00 D22 D23 00 00 00 00
00 00 D24 D25 00 00 00 00
00 00 D26 D27 00 00 00 00
00 00 D28 D29 00 00 00 00
00 00 D30 D31 00 00 00 00 00

```

按上述格式填好数据后,使用 putimage()函数即可将填入缓冲区内的汉字快速的显示到屏幕上指定的位置。在 16 色屏幕状态下可显示出 15 种颜色的汉字,这也是优于汉字操作系统支持下汉字显示的又一突出特点。

三、程序实例

为了使读者容易掌握上述汉字的显示方法,我们编写了下述显示汉字字符串的 Turbo C 程序,供大家参考和使用。该程序可显示灰、红、绿、蓝四种颜色的汉字,其程序清单如下:

```

#include <graphics.h>
#include <stdio.h>
main()
{
char hz[]="西文图形状态下显示汉字";
int mode=VGAHI,adapter=VGA;
int top,left=30,color=0
initgraph(& adapter,&mode,"");
for(color=0;color<=3;color++)
{ /* color=0 灰色
color=1 红色
color=2 绿色
color=3 蓝色 */
top=30+color*16;
hzdisplay(left,top,hz,color);
}
}
/* 在屏幕的指定位置显示汉字串子程序 */
hzdisplay(left,top,chs,color)
int left,top,color;
char chs[];
{
long c1,c2,c3;
int dot[32],i,a;
int hznum,j,left1;
FILE *fi;
char s[134];
if((fi=fopen("cclib","rb"))==NULL)
{
printf("Open CCLIB ERROR!! \n");
exit(0)
}
hznum=strlen(chs)/2;
for (j=0;j<hznum;j++)
{
left1=left+j*16;
c1=chs[j*2];

```



```

c2=chs[j*2+1];
c1=c1&0x7f;
c2=c2&0x7f;
c2=(c2-0x21);
c1=(c1-0x21);
if(c1>=15)c1=c1-7;
c3=(c1*94+c2)*32;
fseek(fi,c3,SEEK_SET);
for(i=0;i<32;i++)
{
fsanf(fi,"%c",&dot[i]);
dot[i]=dot[i]&0x00ff;
s[0]=15;
s[1]=0;
s[2]=15;
s[3]=0;
for(i=4;i<134;i++)s[i]=0;
for(i=0;i<16;i++)for(a=0;a<2;a++)
{
s[i*8+a+4+color*2]=dot[i*2+a];
}
putimage(left1,tops,s,0);
}
fclose(fi);
}

```

本文所述方法也可用于显示 24×24、48×48 点阵的汉字,所提供的源程序稍加改动即可应用。除此之外,该方法还可用于显示用户自己设计的软件标志、图案和各种手写体汉字等,具有广泛的应用价值。

PC Tools 若干高级实用技巧

信阳陆军学院实验中心(464000)李日林

PC Tools 是目前国内非常流行的工具软件,因其界面好、功能强大等突出优点,深受众多用户的喜爱。但许多用户仅用它来修改数据、调试程序,其实,PC Tools 还有一些非常实用的功能,如压缩文件、修复软盘、恢复数据以及对文件或子目录加密等等。本文介绍几则关于 PC Tools 的高级实用技巧,希望对用户的工作有所助益,也希望您介绍一些更好的经验。

一、用 PC Tools 压缩字库文件

五笔字型、CCDOS 等汉字操作系统启动后,汉字库常驻内存,占用了大量的内存空间,使得许多软件不能使用。例如,双软盘五笔字型系统在 PC 机上启动后,C—d BASE III 应用软件因内存空间不够而不能使用。我们知道,汉字库是由一级、二级汉字组成的,其中的二级汉字并不常用,因此,我们可以将字库中二级汉字中的一部分删除,以节约内存空间。例如,将五笔字型汉字库 ZHANG 由 237632 字节压缩至 106560 字

节,内存空间将多出 130K 字节。下面介绍具体方法和步骤。

- 【1】将字库盘插入驱动器 A:
- 【2】C)PCTOOLS/ 启运 PC Tools
- 【3】F3 进入磁盘及特殊功能菜单
- 【4】E 进入编辑状态
- 【5】A/ 指定驱动器 A:,回车
- 【6】F2 选择编辑区
- 【7】R PgDn 进根目录并翻页至有 ZHANG 目录的页面
- 【8】F3 移动光标进入编辑,定位光标于该目录的最后 4 个字节
- 【9】40 A0 03 00→40 A0 01 00 修改文件长度字节

节

- 【10】F5 U 更新操作,确认存盘
- 【11】ESC ESC Y 退出 PCTools

至此,汉字库 ZHANG 由 237632 字节减少为 106560 字节,五笔字型系统在 PC 机上启动后,C—dBASE III 可以使用。

二、用 PCTools 查找忘记文件名的文件

在日常工作中,我们经常会碰到这样的问题:忘记了一个文件的名字,只能记得其中的某些内容或一句话,甚至一个词。在硬盘中存有大量文件的情况下,如果逐个文件翻看的话,将是十分烦琐的。本文介绍一种用 PCTools 快速查找的方法,例如,某文件不知其名字,只记得其中“搞好本年度总结”一句话,下面给出具体的步骤。

- 【1】C)PCTOOLS/ 启动 PCTools
- 【2】F3 进入磁盘及特殊功能菜单
- 【3】E 进入编辑状态
- 【4】C/ 选择 C:盘,回车
- 【5】F 进入文件查找功能
- 【6】在 []ASCII 栏的方括号内填入“搞好本年度总结”,回车,屏幕上将显示该文件的名字
- 【7】ESC ESC Y 退出 PCTools

当然,如果有几个文件都含有这个词,屏幕将显示出几个文件。

三、用 PCTools 隐藏子目录

对子目录进行隐藏是保护文件的一种十分有效的手段。通常的做法是将属性字节改为 12H、13H、16H 及 17H。这样做虽然可以起到一定的保密作用。但在功能强大的 PCTools 面前显得没有什么作用,本文介绍一种用 PCTools 修改子目录长度字节并同时修改属性的方法,能起到更好的隐藏保护作用。查看根目录的文件扇区可以知道,子目录的长度字节通常为 00、00、00、00。通过试验发现,如果将后面的两个字节之一改为非 0 值,即使 PCTools 也不能显示该子目录名。下面以隐藏 C:盘子目录 WIN 为例介绍具体的修改方法。

- 【1】C)PCTools/ 启动 PCTools
- 【2】F3 进入磁盘及特殊功能菜单

- 【3】E 进入编辑状态
- 【4】C↙ 指定 C: 盘, 回车
- 【5】F2 选根目录
- 【6】R PgDn 进入扇区并翻页至子目录 WIN 页面
- 【7】F3 进入编辑
- 【8】移动光标 至 WIN 的属性字节
- 【9】10→12/13/16/17 修改 10 为 12 或 13、16、17
- 【10】移动光标 移动光标至长度字节
- 【11】00→20 将 00 改为 20(或随意指定)
- 【12】F5 确认操作
- 【13】U 更新存盘
- 【14】ESC ESC Y 退出 PCTools

四、用 PCTools 修复软盘

在磁盘的使用过程中,经常遇到下面令人头痛的问题:读盘时,屏幕显示:

```
disk error reading drive A
Abort, Retry, Ignore?
```

按“R”键重试几次仍不能读出文件或数据。通常,我们的做法是将该盘重新格式化,然后再使用,但是盘上的数据也丢失了。其实,这种现象是由于软盘的引导区出错引起的,笔者曾多次用 PCTools 对这类盘的引导区进行恢复,并成功地使盘中的数据完好无损,现将方法介绍给大家。

- 【1】将一片新盘格式化后插入驱动器 A:
- 【2】C>PCTOOLS↙ 启动 PCTools
- 【3】F3 进入磁盘及特殊功能菜单
- 【4】E 进入编辑状态

- 【5】A↙ 指定驱动器 A:, 回车
- 【6】取出好盘,将坏盘插入驱动器 A:
- 【7】F3 编辑
- 【8】F5 更新数据
- 【9】U 确认操作
- 【10】ESC ESC Y 退出 PCTools

至此,恢复完成。如果您的盘上存有大量有用的文件或数据,用这种方法进行修复可以节约大量的时间。

五、用 PCTools 恢复被误删除的文件

对软盘上的文件做删除操作时,偶尔会删掉不该删的文件,如果重新输入该文件,既浪费时间又令人可惜。有关的报刊虽然也介绍过几种恢复文件的方法,如用 DEBUG 调试程序,甚至自编一个专用的程序,但都比较麻烦。本文介绍一种用 PCTools 来恢复文件的方法,操作简单,速度又快,下面以恢复软盘上被误删除的文件 ART.LRL 为例,介绍具体恢复方法和步骤。

- 【1】C>PCTools↙ 进入 PCTools
- 【2】F10 选择路径
- 【3】C↙ 指定驱动器 C:, 回车
- 【4】F3 进入磁盘及特殊功能菜单
- 【5】U 进入恢复状态,回车,回车
此时屏幕显示被删除的文件名,但第一个字符为“?”,扩展名的最后加了一个“@”,本例为?RT.LRL@
- 【6】G 执行操作
- 【7】将“?”改为“A”,回车
- 【8】F1 选择自动进行恢复
- 【9】ESC Y 退出 PCTools

DOS 下图象文件的打印输出

湖南省新晃县公安局(419200) 杨晓慧

金山中文系统以其友好的用户界面和较强的实用性获得了广大用户的青睐。我们在使用其 5.0 版的软系统从 M1570 打印机输出图象文件时,发现图象的斜线和文字的斜笔划均由反向的“毛刺”组成,相当难看。这种“毛刺”与通常所说的笔划“台阶”(低点阵的大号字中比较明显)不同,它甚至使得标准 3 号大小的字也难以辨认。

通过分析金山图象文件的组成结构,我们用 Turbo Pascal 4.0 编制了如下的图象打印程序,并在其集成环境下编译通过,运行成功,解决了在脱离金山系统的情况下打印其图象文件的问题。这很适合不能运行金山软系统的 IBM PC/XT 等微机,当然也可以使没有装金山系统的其他 PC 用户享受到金山图象文件的精美打印效果。编译后所得的可执行文件,既可以在

DOS 下运行,也可由 FoxBASE、dBASE 或其它高级语言程序调用。

程序简介:

本程序由打开文件、获取参数、数据转换和打印 3 个模块组成。

1. 在打开文件模块 open_file 中,用户按提示输入合法的文件名,由程序进行 I/O 检查通过后以读方式打开,考虑数据处理是按字节进行的,故打开文件时元素长度定为 1 字节。

2. 获取参数模块 get_parameters 的功能是通过读文件的有关字节得到图象的宽、高点阵数 w 和 h,并由此计算出要处理的点阵字节总数、1 个打印扫描行字节数,分别用 size_of_data 和 line_buffer 表示。另外还设置了打印机控制码 print_control_code,其中包括单向打印码 uni_direction_print、打印色彩码 print_color、行距设置码 LF_pitch 和点图打印码 bit_image_print。

3. 数据转换和打印模块是整个程序的核心部分,其中包含了转换和打印 2 个过程。本模块依次读取图象文件中按显示顺序排列的点阵数据,反复调用转换和打印过程将其转换成按打印顺序排列的点阵数据打印输出,每次处理 1 个扫描行即 line_buffer 字节的数

据要调用转换和打印过程各 1 次。

在最为关键的转换过程 d2p 中,我们设置了数组常量 y 作为“取点器”,通过逻辑与运算取出图象的象素点(即显示点阵字节中的“1”位)。该过程中的 buf 和 dot 数组分别用来装载显示和打印的点阵数据,其元素个数 6000 是根据本程序给定的最大打印宽度 2000 点×3 确定的,基本上可以满足接近宽行纸宽度的图象打印(此数可由用户根据需要在不超过 2448×3 的范围内改变,但一定要是 3 的倍数)。另外有两点值得注意:①图象文件的点阵数据中,“1”位为亮点(空白点),“0”位才是象素点,这样主要是为了方便显示器。故在用“取点器”取点之前,须将显示数据求反。②笔者从多次实验中证明,打印点阵数据 26 时会引起 M1570 打印机误动作。所以,本过程对值为 26 的打印点阵数据均用 170(2 进制数 10101010)进行逻辑隔位与变换。有趣的是,若对所有的打印点阵数据进行类似的变换,则可以获得不同风格的多种打印效果,当然也包括图象反白。这只需紧跟内层循环加一条对 dot[m]进行相应变换的语句即可。

限于篇幅,有关数据变换的算法原理和打印输出细节就不一一介绍了。

附带说明:

1. 图象文件存盘时必须选择非压缩的 spt 格式。
2. 非 M1570 打印机使用该程序前,要修改 get-parameters 模块中的打印机控制码和打印数据量说明方式。
3. 对于打印点阵数据要求高位在上的打印机,应将 d2p 过程中带下划线的部分(即 8-d)换为 d-1。

程序如下:

```
program dyspt;
var
  wjm:string[20];
  f:file;
  w,h,rcs,size_of_data:word;
  line_buffer,i,j:integer;
  k:longint;
  lst:text;
  uni_direction_print:string;
  print_color:string;
  LF_pitch:string;
  bit_image_print:string;
  print_control_code:string;
  dot_col_num:string;
  z:array[1..4] of byte;

  procedure open_file;
  begin
  repeat
    write('#? Please enter the name of image file:');
    readln(wjm);
    [ $I+ ]
    assign(f,wjm);    assign(lst,'lpt1');
    reset(f,1);      rewrite(lst);
```

```
    [ $I+ ]
until Ioresult=0;
end;

procedure get_parameters;
begin
seek(f,34);
blockread(f,w,2,rcs);
blockread(f,h,2,rcs);
line_buffer:=w*3;
size_of_data:=w*h div 8;
seek(f,64);
i:=0;
rcs:=0;
z[1]:=w mod 10+48;
z[2]:=w mod 100 div 10+48;
z[3]:=w mod 1000 div 100+48;
z[4]:=w mod 10000 div 1000+48;
uni_direction_print:=chr(27)+chr(62);
print_color:=chr(27)+'C4';
LF_pitch:=chr(27)+'T48';
bit_image_print:=chr(27)+'G';
print_control_code:=uni_direction_print+print_color;
or;
print_control_code:=print_control_code+LF_pitch;
print_control_code:=print_control_code+bit_image_print;
dot_col_num:=chr(z[4])+chr(z[3])+chr(z[2])+chr(z[1]);
end;

procedeuure convert_and_print(qc:word);
const
  y:array[1..8] of byte=(128,64,32,16,8,4,2,1);
var
  buf,dot:array[1..6000] of byte;
  x:array[1..8] of byte;

  procedure d2p;
  var
    a,b,c,d:byte;
    m,n:word;
  begin
    for j:=1 to qc do
      dot[j]:=0;
    for a:=1 to w div 8 do
      for b:=1 to 8 do
        for c:=1 to 3 do
          begin
            for j:=1 to 8 do
              x[j]:=0;
            m:=24*(a-1)+3*(b-1)+c;
            for d:=1 to 8 do
              begin
                n:=((c-1)*8+d-1)*w div 8+a;
                x[d]:=y[b] and (not buf[n]);
```



```

x[d]:=x[d] shl (b-1) shr (8-d);
dot[m]:=dot[m] or x[d];
end;
if dot[m]=26 then
dot[m]:=dot[m] and 170;
end;
end;
end;

procedure prt;
begin
write(lst,print_control_code+dot_col_num);
for j:=1 to qc do
write(lst,chr(dot[j]));
write(lst,chr(13)+chr(10));
end;

begin
repeat
for j:=1 to qc do
buf[j]:=255;
blockread(f,buf,qc,rcs);
if rcs<=qc then
begin
d2p;
prt;
i:=i+1;
end;
until eof(f);
close(f); close(lst);
end;

begin
open_file;
get_parameters;
convert_and_print(line_buffer);
end.

```

如何使 PC 机具有实时时钟

重庆航天工业公司(630039) 李维宪

早期的 IBM-PC 系列计算机由于没有保存机器配置的 CMOS 电路,也没有保持时钟的电路,因此不可能具有机器的实时钟,机器关机或热启动后,时间就丢失了。为了能够像 286 以上档次的机器那样可以保持时间准确,现在大部分 PC 机都使用了多功能卡,这种功能卡除了软盘接口、打印机接口、串行口、游戏口以外,还有一个实时钟接口,且这类卡的一个最明显特征是卡上安装有一个五分硬币大小的 3 伏纽扣电池,这个电池是为了在机器关机以后仍然能够让多功能卡上的计时电路工作而提供的能源,有了这也就具备了

实现 PC 机实时钟的条件。但是,这个实时钟又是必须由一个时钟程序来驱动的,据了解,很多用户由于找不到时钟驱动程序,没法使用实时钟,只好每次机器启动后人工输入当前日期和时间,除显得比较繁琐,给工作带来不便以外,计算机的资源也没有得到充分利用。为此,笔者编制了一个获取和转换机器实时钟的程序,介绍给计算机同行,希望对您有所帮助。程序分(1)和(2),前者读取卡上的时间并转换为机器时间,后者将当前日期和时间显示并将其存入卡中。

【使用方法】两个程序可以同时放在 AUTOEXEC. BAT 文件内执行,注意:执行顺序,程序(1)必须在前,程序(2)在后,修改时间可以用 DOS 下的 DATE 和 TIME 命令,然后执行一次程序(2),则将修改后的时间保存起来了,以后机器启动就在新的时钟下工作,在板上电池完好的情况下,您就可以不必为时间不准而费心了。

【运行环境】IBM PC-XT 及其兼容机, DOS 2.0 以上各种版本操作系统。

在 DEBUG 下用 A100 命令键入以下程序:

(1)获取时间程序 GET-CLK.COM 文件长度

```

56H
0100 BA4402 MOV DX,0244
0103 E83900 CALL 013F
0106 88C5 MOV CH,AL
0108 E83400 CALL 013F
010B 88C1 MOV CL,AL
010D E82F00 CALL 013F
0110 88C7 MOV BH,AL
0112 E82A00 CALL 013F
0115 88C2 MOV DL,AL
0177 88FE MOV DH,BH
0119 B42D MOV AH,2D
011B CD21 INT 21
011D BA4702 MOV DX,0247
0120 E81C00 CALL 013F
0123 88C5 MOV CH,AL
0125 E81700 CALL 013F
0128 88C1 MOV CL,AL
012A 83C04 ADD DX,+04
012D 31C0 XOR AX,AX
012F E80D00 CALL 013F
0132 05BC07 ADD AX,07BC
0135 87D1 XCHG DX,CX
0137 89C1 MOV CX,AX
0139 B42B MOV AH,2B
013B CD21 INT 21
013D CD20 INT 20
013F EC IN AL,DX
0140 51 PUSH CX
0141 50 PUSH AX
0142 B90400 MOV CX,0004
0145 D2E8 SHR AL,CL
0147 88C1 MOV CL,AL
0149 B80600 MOV AX,0006

```


丢失,代之以汉字系统文件。这种损失是很大的。

那么怎样解决二级汉字的问题呢?在一级汉字驻留内存的情况下,容易想到:可否将一级汉字去掉,生成一个单独的二级汉字库文件呢?这样的话,系统二号盘是能装下的,且富余的空间是够系统使用的(FoxBASE+系统在工作时,系统盘需一定空间供其使用)。实现这一方案存在两个问题:一是放在不同盘上的字库,系统能否正确判读的问题;二是如何获得单独的二级汉字库的问题。将完整的“CCLIB”文件放于工作盘上,没有特殊处理是不能被正确判读的。通过观察分析,笔者发现启动汉字系统,调入一级汉字时,系统自动记忆二级汉字所在的驱动器和磁盘扇区,以后直接到该驱动器磁盘的相应扇区查找,不是从头查找。也就是说,要实现这一方案,还必须保证这个单独的二级汉字库与 CCDOS 系统盘上的汉字库存放的位置不能有丝毫的出入。要达到这一点是不容易的。笔者通过反复实践发现一种方法,既能保证成功,操作也很简单。现提供给大家,供同行参考。

首先格式化两张磁盘(其中不能有坏道和扇区),均用 COPY 命令拷入“CCLIB”文件。取其中一张再拷入 CCDOS 的其它文件,而成为 CCDOS 系统盘;取另一张盘,在 PCTools 中进入磁盘查询编辑状态,翻页到第五逻辑扇区。这时看到的是“CCLIB”文件的目录项,其第 26、27 字节的文件起始簇号为“0200”,此时按 F3 进入编辑状态,将第 26、27 字节改为“9000”,按 F5 存盘退出。

如果不用 PCTools 而使用 DEBUG,则将含 DEBUG 的磁盘放于 A 驱,仅含字库的盘放于 B 驱,执行:

```
A>DEBUG
-L CS:100 1 5 1
-E CS:11A
× × × ×:011A 02.90
-W CS:100 1 5 1
-Q
```

大家知道,经这样修改的磁盘,仅改变了 CCLIB 文件的起始簇号,还存在文件长度的修改和磁盘文件分配表的修改问题。这两项的计算修改是比较麻烦的,特别是磁盘 FAT 表的修改。最简单的方法是利用 CHKDSK 命令对磁盘检查修整。方法是,将含 CHKDSK 文件的盘放于 A 驱,字库盘放于 B 驱,执行:A>CHKDSK B:/F/V 命令,待显示:

```
“142 Lost clusters found in 1 chains.
Convert lost chains to files (Y/N)?”
```

时键入“Y”并回车,结束后删去盘上的“FILE0000.CHK”文件。

然后,再将 FOXPLUS.RSC、FOXPLUS.OVL、COMMAND.COM 文件拷入该盘。这时,磁盘还剩下 73728 字节的空,足够系统使用。

使用时,先在 A 驱启动汉字系统,调入一级汉字,再进入 FOX 系统;A 驱放 FOX 的 2# 盘,B 驱放任意

工作盘,方便实用,不再出现前面所说诸问题。

注:该过程使用 DOS 3.3、CCDOS 4.0、FoxBASE +2.1、PCTools 5.0;该方法适用于双软驱 IBM PC/XT 及其兼容机。

利用电脑预测儿童青少年成人时的身高

泸州医学院(646000)黎彦才 米绍平 徐奎

儿童的生长发育是社会普遍关心的问题。而身高预测是儿童生长发育预测的一种,它对体育、文艺部门的选材、内分泌疾病疗效的观察等都有较大的实用价值。家长们对自己的孩子以后能长多高,是否长得太高或太矮十分关心。据报道,成人身高有 75% 取决于遗传,仅有 25% 取决于环境。即遗传是决定身高的潜力,而潜力能否充分发挥,则有赖于良好的环境条件,如营养、体育锻炼等等。身高的生长发育具有很大的稳定性,也为身高的预测创造了良好的条件。利用儿童少年时间生长发育的规律,结合考虑多种遗传及环境因素对发育之影响,经过大量数据调查和对人体测量资料的统计学处理,可以编制成多种公式,对儿童、少年的成年时的身高进行预测。

本程序根据《医学百科全书、儿童少年卫生学》提供的公式和资料,可以十分方便地对儿童、少年成人时的身高进行预测。其提供的预测方法有三种:

一、利用父母身高进行预测。公式为:

儿子成人时的身高=(父高+母高)×0.54

女儿成人时的身高=(父高×0.923+母高)/2

二、利用儿童当时的年龄和身高进行预测:

$H_m = A + (B \times C)$

其中: H_m 为儿童成人时的身高,A 为常数,B 为相应系数,C 为当时的身高。A、B 是通过统计资料制成的表中查得,这里可参阅《医学百科全书,儿童少年卫生学》第 36 页预测身高表。本法适用于 2 岁半至 20 岁的儿童和青少年。

三、利用足长进行预测:

中国青少儿体质调查组根据 1979 年对 11 万名城 7—25 岁男女学生测试数据研究出利用足长预测儿童成人时身高的方法,并制定了相应预测表。根据儿童、青少年的年龄和足长查预测表即可查得其成人时的身高(参阅《中国医学百科全书,儿童少年卫生学》P36

10 REM 儿童身高预测程序,参阅《中国医学百科全书,儿童少年卫生学分册》,P36.

20 DIM A1(19),B1(19),A2(19),B2(19),A(65,10),B(52,9)

40 REM 方法 2 所用的 A,B 值

50 RESTORE 600;FOR I=1 TO 19

60 READ A1(I),B1(I),A2(I),B2(I);NEXT I

70 REM 方法 3 所用的根据足长(I)和年龄(J)确定身高数据

```

80 RESTORE 900;FOR I=1 TO 65;FOR J=1 TO 10
90 READ A(I,J);NEXT J;NEXT I
100 RESTORE 1600;FOR I=1 TO 52;FOR J=1 TO 9
110 READ B(I,J);NEXT J;NEXT I
150 CLS;LOCATE 1,20;PRINT "儿童身高预测"
160 PRINT "功能:根据遗传因素和儿童生长发育状况预测
    儿童成人时的身高。";PRINT
170 INPUT "姓名/编号(结束敲 RETURN):",XM¥
180 IF XM¥="" THEN END
190 PRINT "性别:1-城男,2-城女,3-农男,4-农女:";
200 INPUT " ",XB;IF XB(1 OR XB)4 THEN 190
210 INPUT "现在年龄(岁,个月):",N,YS
220 INPUT "父亲身高(cm):",FG
230 INPUT "母亲身高(cm):",MG
240 INPUT "现在身高(cm):",SG
250 IF N)=7 THEN INPUT "现在足长(cm):",ZC
300 REM 方法 1:公式计算
310 IF FG=0 OR MG=0 THEN H1=0;GOTO 350
320 IF XB=1 OR XB=3 THEN H1=(FG+MG)/2*1.08
330 IF XB=2 OR XB=4 THEN H1=(FG*.923+MG)/2
350 REM 方法 2:未来身高=A+(B×C),A:常数,B:相应系
    数,C:当时年龄的身高,本法误差≤3cm.
360 IF YS=>6 THEN S=N+.5 ELSE S=(N-1)+.5
370 IF S<2.5 OR S>20.5 OR XG=0 THEN H2=0;GOTO
400 ELSE I=INT(S-1.5)
380 IF XB=1 OR XB=3 THEN H2=A1(I)+B1(I)*SG
    ELSE H2=A2(I)+B2(I)*SG
400 REM 方法 3:据足长查表求身高
410 IF XB=3 OR XB=4 THEN 500
420 IF XB=2 THEN 470
430 IF N<7 OR N>16 THEN HE=0;GOTO 500
440 IF ZC<17 OR ZC>29.8 THEN H3=0;GOTO 500
450 I=INT((ZC-16.8)*5);J=N-6;H3=A(I,J);'男
460 GOTO 500
470 IF N<7 OR N>15 THEN H3=0;GOTO 500
480 IF ZC<17.2 OR ZC>27.4 THEN H3=0;GOTO 500
490 I=INT((ZC-17)*5);J=N-6;H3=B(I,J);'女
500 REM 比较,判断,打印
510 H=H1;IF H<H2 THEN H=H2
520 IF H<H3 THEN H=H3
530 PRINT;IF H=0 THEN 560
540 PRINT "你未来的身高大约是:";USING "###";H;
550 PRINT "cm.";PRINT;PRINT TAB(5);;GOTO 570
560 PRINT "缺数据,不能预测,";
570 PRINT "敲任一键...";K¥=INPUT¥(1);GOTO 150
600 REM 方法 2 所用预测身高表
610 REM 男 A,B 女 A,B 年龄
620 DATA 86.9,1.02,99.75,0.75;'2.5 岁
630 DATA 76.76,1.02,86.71,0.81;'3.5 岁
640 DATA 76,0.97,73.04,0.88;'4.5 岁
650 DATA 75.44,0.91,52.22,1.01;'5.5 岁
660 DATA 73.09,0.88,50.09,0.97;'6.5 岁
670 DATA 71.85,0.85,51.68,0.91;'7.5 岁
680 DATA 70.89,0.82,54.57,0.85;'8.5 岁
690 DATA 71.86,0.78,68.63,0.71;'9.5 岁
700 DATA 71.87,0.75,90.89,0.52;'10.5 岁
710 DATA 75.38,0.70,87.94,0.52;'11.5 岁
720 DATA 98.97,0.52,77.08,0.57;'12.5 岁
730 DATA 111.68,0.42,37.41,0.8;'13.5 岁
740 DATA 100.38,0.47,12.4,0.94;'14.5 岁
750 DATA 68.02,0.64,6.57,0.97;'15.5 岁
760 DATA 34.11,0.82,4.39,0.98;'16.5 岁
770 DATA 15.85,0.92,2.15,0.99;'17.5 岁
780 DATA 6.13,0.97,1.71,0.99;'18.5 岁
790 DATA 2,0.99,0.1;'19.5 岁
800 DATA 2,0.99,0.1;'20.5 岁
900 REM 男子足长预测身高表,单位:cm
910 REM 7 岁,8 岁,9 岁,10 岁,11 岁,12 岁,13 岁,14 岁,15
    岁,16 岁;'足长
920 DATA 157,152,146,0,0,0,0,0,0;'17
930 DATA 159,154,147,0,0,0,0,0,0;'17.2
940 DATA 160,155,149,0,0,0,0,0,0;'17.4
950 DATA 162,157,151,145,0,0,0,0,0;'17.6
960 DATA 164,159,153,147,0,0,0,0,0;'17.8
970 DATA 166,161,154,148,0,0,0,0,0;'18
980 DATA 168,163,156,150,145,0,0,0,0;'18.2
990 DATA 170,164,158,152,147,0,0,0,0;'18.4
1000 DATA 172,166,159,153,148,0,0,0,0;'18.6
1010 DATA 173,168,161,155,150,0,0,0,0;'18.8
1020 DATA 175,170,163,157,151,145,0,0,0;'19
1030 DATA 177,171,165,158,153,147,0,0,0;'19.2
1040 DATA 179,173,166,160,155,148,0,0,0;'19.4
1050 DATA 181,175,168,162,156,150,0,0,0;'19.6
1060 DATA 183,177,170,163,158,151,146,0,0,0;'19.8
1070 DATA 184,179,171,165,159,153,147,0,0,0;'20
1080 DATA 186,180,173,166,161,155,149,0,0,0;'20.2
1090 DATA 188,182,175,168,163,156,150,145,0,0;'20.4
1100 DATA 190,184,177,170,164,158,151,146,0,0;'20.6
1110 DATA 192,186,178,171,166,159,153,148,145,0;'
    20.8
1120 DATA 194,188,180,173,167,161,154,149,146,145;'
    21
1130 DATA 195,189,182,175,169,162,156,151,148,146;'
    21.2
1140 DATA 197,191,183,176,171,164,157,152,149,148;'
    21.4
1150 DATA 199,193,185,178,172,165,159,154,150,149;'
    21.6
1160 DATA 201,195,187,180,174,167,160,155,152,151;'
    21.8
1170 DATA 203,196,189,181,175,168,162,156,153,152;'
    22
1180 DATA 205,198,190,183,177,170,163,158,155,153;'
    22.2
1190 DATA 0,200,192,185,179,171,165,159,157,155;'
    22.4
1200 DATA 0,202,194,186,180,173,166,161,157,156;'
    22.6
1210 DATA 0,204,195,188,182,174,168,162,159,157;'
    22.8
1220 DATA 0,205,197,190,183,176,169,163,160,159;'23
1230 DATA 0,0,199,191,185,177,171,165,162,160;'23.2

```


1240 DATA 0,0,201,193,187,179,172,166,163,162: '23.4
 1250 DATA 0,0,202,194,188,181,174,168,164,163: '23.6
 1260 DATA 0,0,204,196,190,182,175,169,166,165: '23.8
 1270 DATA 0,0,206,198,191,184,176,171,167,166: '24
 1280 DATA 0,0,0,199,193,185,178,172,169,167: '24.2
 1290 DATA 0,0,0,201,195,187,179,173,170,169: '24.4
 1300 DATA 0,0,0,203,196,188,181,175,171,170: '24.6
 1310 DATA 0,0,0,204,198,190,182,176,173,171: '24.8
 1320 DATA 0,0,0,206,199,191,184,178,174,173: '25
 1330 DATA 0,0,0,201,193,185,179,175,174: '25.2
 1340 DATA 0,0,0,0,202,194,187,181,177,175: '25.4
 1350 DATA 0,0,0,0,204,196,188,182,178,177: '25.6
 1360 DATA 0,0,0,0,206,197,190,183,180,178: '25.8
 1370 DATA 0,0,0,0,0,199,191,185,181,180: '26
 1380 DATA 0,0,0,0,0,200,193,186,182,181: '26.2
 1390 DATA 0,0,0,0,0,202,194,188,184,182: '26.4
 1400 DATA 0,0,0,0,0,203,196,189,185,184: '26.6
 1410 DATA 0,0,0,0,0,205,197,190,187,185: '26.8
 1420 DATA 0,0,0,0,0,199,192,188,186: '27
 1430 DATA 0,0,0,0,0,200,193,189,188: '27.2
 1440 DATA 0,0,0,0,0,201,195,191,189: '27.4
 1450 DATA 0,0,0,0,0,203,196,192,191: '27.6
 1460 DATA 0,0,0,0,0,0,204,198,194,192: '27.8
 1470 DATA 0,0,0,0,0,0,206,199,195,193: '28
 1480 DATA 0,0,0,0,0,0,200,196,195: '28.2
 1490 DATA 0,0,0,0,0,0,202,198,196: '28.4
 1500 DATA 0,0,0,0,0,0,203,199,198: '28.6
 1510 DATA 0,0,0,0,0,0,205,201,199: '28.8
 1520 DATA 0,0,0,0,0,0,202,200: '29
 1530 DATA 0,0,0,0,0,0,203,202: '29.2
 1540 DATA 0,0,0,0,0,0,205,203: '29.4
 1550 DATA 0,0,0,0,0,0,0,206: '29.6
 1560 DATA 0,0,0,0,0,0,0,204: '29.8
 1600 REM 女子足长预测身高表,单位:cm
 1610 REM 7岁,8岁,9岁,10岁,11岁,12岁,13岁,14岁,
 15岁: '足长
 1620 DATA 151,145,140,134,0,0,0,0,0: '17.2
 1630 DATA 153,147,141,136,0,0,0,0,0: '17.4
 1640 DATA 155,149,143,137,0,0,0,0,0: '17.6
 1650 DATA 156,150,144,139,0,0,0,0,0: '17.8
 1660 DATA 158,152,146,140,136,0,0,0,0: '18
 1670 DATA 160,154,148,142,137,0,0,0,0: '18.2
 1680 DATA 162,156,149,143,139,0,0,0,0: '18.4
 1690 DATA 163,157,151,145,140,135,0,0,0: '18.6
 1700 DATA 165,159,152,147,142,137,0,0,0: '18.8
 1710 DATA 167,161,154,148,143,138,136,0,0: '19
 1720 DATA 169,162,156,150,145,139,137,136,0: '19.2
 1730 DATA 170,164,157,151,146,141,138,137,136: '19.4
 1740 DATA 172,166,159,153,148,142,140,139,137: '19.6
 1750 DATA 174,167,161,154,149,144,141,140,139: '19.8
 1760 DATA 176,169,162,156,151,145,143,141,140: '20
 1770 DATA 177,171,164,157,152,147,144,143,142: '20.2
 1780 DATA 179,172,165,159,154,148,146,144,143: '20.4
 1790 DATA 181,174,167,161,155,150,147,146,144: '20.6
 1800 DATA 183,176,169,162,157,151,148,147,146: '20.8
 1810 DATA 184,178,170,164,158,153,150,148,147: '21

1820 DATA 186,179,172,165,160,154,151,150,149: '21.2
 1830 DATA 188,181,174,167,161,155,153,151,150: '21.4
 1840 DATA 190,183,175,168,163,157,154,153,151: '21.6
 1850 DATA 191,184,177,170,164,158,155,154,153: '21.8
 1860 DATA 193,186,178,171,166,160,157,156,154: '22
 1870 DATA 195,188,180,173,167,161,158,157,156: '22.2
 1880 DATA 0,189,182,175,169,163,160,158,157: '22.4
 1890 DATA 0,191,183,176,170,164,161,160,158: '22.6
 1900 DATA 0,193,185,178,172,166,163,161,160: '22.8
 1910 DATA 0,194,187,179,173,167,164,163,161: '23
 1920 DATA 0,196,188,181,175,168,165,164,163: '23.4
 1930 DATA 0,0,190,182,176,170,167,165,164: '23.4
 1940 DATA 0,0,191,184,178,171,168,167,165: '23.6
 1950 DATA 0,0,193,185,179,173,170,168,167: '23.8
 1960 DATA 0,0,195,187,181,174,171,170,168: '24
 1970 DATA 0,0,0,189,182,176,173,171,170: '24.2
 1980 DATA 0,0,0,190,184,177,174,172,171: '24.4
 1990 DATA 0,0,0,192,185,179,175,174,172: '25.6
 2000 DATA 0,0,0,193,187,180,177,175,174: '24.8
 2010 DATA 0,0,0,195,188,182,178,177,175: '25
 2020 DATA 0,0,0,0,190,183,180,178,177: '25.2
 2030 DATA 0,0,0,0,0,191,184,181,180,178: '25.4
 2040 DATA 0,0,0,0,0,193,186,183,181,179: '25.6
 2050 DATA 0,0,0,0,194,187,184,182,181: '25.8
 2060 DATA 0,0,0,0,196,189,185,184,182: '26
 2070 DATA 0,0,0,0,0,190,187,185,184: '26.2
 2080 DATA 0,0,0,0,0,192,188,187,185: '26.4
 2090 DATA 0,0,0,0,0,193,190,188,186: '26.6
 2100 DATA 0,0,0,0,0,195,191,189,188: '26.8
 2110 DATA 0,0,0,0,0,193,191,189: '27
 2120 DATA 0,0,0,0,0,194,192,191: '27.2
 2130 DATA 0,0,0,0,0,0,195,194,192: '27.4

五笔字型系统 用压缩字库

潍坊造纸总厂(261041) 徐庆恩

五笔字型汉字操作系统以其输入汉字重码率低、输入速度快而深受广大用户的喜爱,是很流行的汉字操作系统。

五笔字型系统的汉字库占系统内存 240K 左右。对于内存为 512K 的 PC 机及其兼容机来说运行大点的程序如 dBASE III、FoxBASE 等就出现内存不够,甚至无法运行的问题。

目前好些用户使用 2.13 或 CCDOS 4.0 等汉字操作系统采用汉字库部分驻留内存,部分存于硬盘的办法。尽管解决了内存不足的问题,但也有无法弥补的缺点。由于汉字库未驻留内存部分在硬盘里,读这部分汉字时,需频繁地读硬盘,使程序的运行速度大大地降

低。

中科院希望公司的 UC DOS 的压缩字库是较理想的汉字库,仅占内存 64K 左右。但是习惯于五笔字型的用户又不愿意使用 UC DOS。对此笔者成功地解决了五笔字型汉字操作系统 V2.00 用 UC DOS 的压缩字库,使两者的优点结合在一起。现将原理和解决的办法介绍给同行共享。

UC DOS 的压缩字库名为 HLIB. EXE (文件长 65825 字节)。其工作原理是:读压缩字库驻留内存,并设置 INT 6AH 软中断,计算汉字字模地址。该中断的入口参数:AX=两字节的汉字国标码。其出口参数:ES:0 为所查汉字的 16×16 点阵的字模起始地址。

五笔字型汉字系统读汉字库 CCLIB 和计算汉字字模地址的子程序在 WB2. EXE (或名为 ZHENG. EXE)的程序里。读汉字库 CCLIB 的部分,因用压缩字库 HLIB. EXE 而必须跳过,应在 CS:50EAH 处汇编一条指令 JMP 5129H 跳过去。计算汉字字模地址的原子程序如图②。根据 INT 6AH 软中断的入口和出口参数来修改该子程序,这是关键。先将 DX 中的汉字机内码转换为国标码,并送入 AX,调用 INT6AH 软中断,把返回的 ES 内的值送 DX,最后处理出栈的数据。修改好的计算汉字字模地址子程序如图①。

修改的前提是将 WB2. EXE 文件改名为 WB2 非 EXE 文件。图①②皆是从 WB2 文件反汇编出的。改完后再改名为 WB2. EXE。

与之相配的汉字打印驱动程序也必须修改。该程序计算字模地址子程序与图②基本相同,把该子程序修改为如图③。其地址也是改名为非 EXE 文件的地址。注意:图③是 1724P16 打印驱动程序反汇编出的。其它打印驱动程序该子程序的地址是不同的。下表给出了部分打印驱动程序计算字模地址子程序的地址。

文件名	修改地址	文件长
LQ1500D	2C28H	11264
ALL9P	2312H	8848
ALL24P	CA4H	3584
1724P	C62H	3456
LQ1500	C93H	3584
BROTHER	2C28H	11264
M24P	CA4H	3478
H3070	3CA3H	20460
M1570	E4DH *	3584

注表中带 * 者修改子程序略有不同。

使用方法:建一批处理文件其内容如下:

```
WBHZ
WBCH
WB1
HLIB
WB2
1724P16
```

五笔字型汉字操作系统用 UC DOS 压缩字库是在

IBM PC/XT、286 机上调试通过的。

```
-u29f2 2a04
37AC:29F2 81E27F7F  AND    DX,7F7F
37AC:29F6 50        PUSH   AX
37AC:29F7 06        PUSH   ES
37AC:29F8 89D0     MOV    AX,DX
37AC:29FA CD6A     INT    6A
37AC:29FC 8CC2     MOV    DX,ES
37AC:29FE 07        POP    ES
37AC:29FF 58        POP    AX
37AC:2A00 1F        POP    DS
37AC:2A01 59        POP    CX
37AC:2A02 5B        POP    BX
37AC:2A03 5F        POP    DI
37AC:2A04 C3        RET
```

图 1

```
-u29f2 2A20
37AC:29F2 81E27F7F  AND    DX,7F7F
37AC:29F6 EB05     JMP    29FD
37AC:29F8 297502    SUB    [DI+02],SI
37AC:29FB B628     MOV    DH,28
37AC:29FD 80FE30    CMP    DH,30
37AC:2A00 7203     JB     2A05
37AC:2A02 80EE05    SUB    DH,05
37AC:2A05 80EE21    SUB    DH,21
37AC:2A08 B05E     MOV    AL,5E
37AC:2A0A F6E6     MUL   DH
37AC:2A0C 80EA21    SUB    DL,21
37AC:2A0F 32F6     XOR   DH,DH
37AC:2A11 03C2     ADD   AX,DX
37AC:2A13 D1E0     SHL   AX,1
37AC:2A15 1F        POP    DS
37AC:2A16 2E        CS:
37AC:2A17 03067500  ADD   AX,[0075]
37AC:2A1B 8BD0     MOV   DX,AX
37AC:2A1D 59        POP   CX
37AC:2A1E 5B        POP   BX
37AC:2A1F 5F        POP   DI
37AC:2A20 C3        RET
```

图 2

```
-U2C28 2C36
37AC:2C28 81E27F7F  AND    DX,7F7F
37AC:2C2C 50        PUSH   AX
37AC:2C2D 06        PUSH   ES
37AC:2C2E 89D0     MOV    AX,DX
37AC:2C30 CD6A     INT    6A
37AC:2C32 8CC2     MOV    DX,ES
37AC:2C34 07        POP    ES
37AC:2C35 58        POP    AX
37AC:2C36 C3        RET
```

图 3

TRAVELLER(C)病毒的发现与清除

北京石化工程公司计算中心
(100029) 桂宁

TRAVELLER(C)病毒是新近开始流行的一种新病毒,具有极强的传染性和隐蔽性,用公安部的81种病毒检测程序SCAN.EXE检测不到该病毒。本人通过分析,找到了发现和消除该病毒的方法。

一、TRAVELLER(C)病毒的表现形式

TRAVELLER(C)病毒是一种寄生性病毒,其宿主是可执行文件,即.COM和.EXE文件,其表现形式具有如下特点:

1. 受感染的.COM文件长度增加1220个字节,.EXE文件的长度增加1223个字节。
2. 运行的速度减慢。
3. 在一定条件下,屏幕显示“!!!!!! ---》Traveller(c) BUPT 1991. 4 Don't panic I'm harmless《---!!!!!!”。

二、TRAVELLER(C)病毒的传染方式

TRAVELLER(C)病毒通过文件进行传染。当运行带有病毒的文件时,病毒驻留内存,此后,只要一运行.COM和.EXE文件,这些文件就会感染上病毒。

三、TRAVELLER(C)病毒的运行机制

当运行带该种病毒的文件时,首先执行病毒程序部分,判断内存是否驻留有病毒程序,若有则恢复原文件的有关数据,转去执行原文件,否则修改INT 21H中断入口地址使之指向病毒的相应中断服务程序,驻留后恢复原文件有关数据,转去执行原文件。

占据INT 21H入口的病毒程序拦截了INT 21H功能调用4BH、36H,对文件实行传染后转入正常INT 21H入口,其它功能调用转病毒发作的条件部分,若满足条件则病毒表现部分发作,在屏幕上显示一行信息,否则转入正常的INT 21H的入口。

四、TRAVELLER(C)病毒的发现

1. 内存病毒检测

用DEBUG检查INT 21H的中断向量。

C)DEBUG

—D0:84

如果0000:84—87单元里的内容为E1 01 XX XX,则内存一定已有该种病毒。

2. 文件病毒检测

如果发现.COM文件增加1220字节,.EXE文件增加1223字节,则表明该文件已感染上该病毒。

也可用DEBUG来检测

C)DEBUG * * *.COM

—U

JMP XXXX

.....
—T
JMP XXXX
.....

—T
MOV AL,43
CALL XXXX

C)DEBUG * * *.EXE

—U
MOV AL,45
CALL XXXX

即可确认为该种病毒。

五、TRAVELLER(C)病毒的消除

首先使系统不带毒或用不带毒的系统盘启动。

1. .COM文件病毒的消除

DEBUG * * *.COM

—D101

取前2个字节XXXX加3再与11AH相加作为地址,设为YYYY。

—DYyyy

记录前3个字节

—E100

用记录的数修改前3个字节

—RCX

文件长度减掉4C4H个字节。

—W

—Q

例如:EXAMPLE ONE

2. .EXE文件病毒的消除

C)DEBUG * * *.EXE

—DCS:100

记下10B—10C,10D—10E,107—108,109—10A单元的值。

—Q

C)REN * * *.EXE * * *

C)DEBUG * * *

—D100

修改文件头:

(a)用上面记下的10B—10C的值修改10E—10F的值。

(b)用上面记下的10D—10E的值修改110—111的值。

(c)用上面记下的107—108的值修改114—115的值。

(d)用上面记下的109—10A的值修改116—117的值。

—E102

修改文件长度:

(文件页长度-1)X200H+文件映象长度-4C7H为原文件长度,转换成页长度和映象长度后修改102—105处值。

```

—RCX
文件长度减 4C7H 字节。
—W
—Q
例如:EXAMPLE TWO
EXAMPLE ONE
C>DEBUG COMMAND.COM
—R
AX=0000 BX=0000 CX=6780 DX=0000 SP=FFFE .....
DS=13E1 ES=13E1 SS=13E1 CS=13E1 IP=0100 .....
13E1:0100 E9B962 JMP 63BC
—D101
13E1:0100 B9 62 BA DA 0A 3D 05—00 74 1B BA BF 0A 3D
02
13E1:0110 00 74 13 BA 85 0A 3D 08—00 74 0B BA 71 0A
3D 0B
.....
—H62B9 3
62BC 62B6
—H62BC 11A
63D6 61A2
—D63D6
13E1:63D0 E9 2D—0D B0 43 E8 35 00 FC BF
13E1:63E0 00 01 BE 1A 01 03 F3 B9—03 00 F3 A4 B8 00 01
50
.....
—E100
13E1:0100 E9. B9. 2D 62. 0D
—RCX
CX 6780
:
—H6780 4C4
6C44 62BC
—RCX
CX 6780
:62BC
—W
Writing 62BC bytes
—Q
EXAMPLE TWO
C>DEBUG GWBASIC.EXE
—R
AX=0000 BX=0001 CX=1867 DX=0000 SP=0A10 .....
DS=13E1 ES=13E1 SS=252B CS=251B IP=0138 .....
251B:0138 B045 MOV AL,45
—DCS:100
251B:0100 E9 9D 15 EB IB 2E 7D A4—FA 98 00 3A 11 80
00 60
251B:0110 14 4B 02 2A 2E 2A 00 43—4F 4D 45 58 45 E9 2D
0D
.....
—Q
C>REN GWBASIC.EXE GW
C>DEBUG GW
—D100

```

```

13C6:0100 4D 5A 67 00 8E 00 61 00—20 00 08 00 FF FF 3A
11
13C6:0110 10 0A B7 59 38 01 2A 11—20 00 00 00 F3 10 00
05
.....
—E10E
13C6:010E 3A. 11.
13C6:0110 10. 80 0A. 00
—E114
13C6:0114 38. A4 01. FA 2A. 98 11. 00
—E102
13C6:0102 67. A0 00. 01 8E. 8B
—RCX
CX 1A67
:
—H1A67 4C7
1F2E 15A0
—RCX ]
CX 1A67
:15A0
—W
Writing 115A0 bytes
—Q
C>REN GW GWBASIC.EXE

```

CPAV—— 使您的电脑百毒不侵

深圳大学电脑协会软件部
(518080) 郭文典

CPAV 是近来出名的解毒工具,是新版 PCTools 的附件,与著名的 SCAN 相比,CPAV 具有易用、集成化及可扩充等特点。

CPAV 主文件的下拉菜单(或条形菜单)方便友好,并支持鼠标,具有多种功能:1)侦察:针对各种病毒的通常感染部位及病毒特征进行快速检查(也可详细检查),然后,在每个目录下产生 CHECKSUM 文件,保留可执行文件大小等资料,若已有该文件存在则对比当前文件资料,有异则报警(病毒往往会改变文件大小、日期等)。2)清除:对一般遭感染文件,可选择清除、删除、备份及不动作等处理。3)免疫:CPAV 将会按要求在可执行文件后加一小段(约 1K)的代码,使程序运行时能自我检查,若遭感染则报警。

除上述功能外,CPAV 还有详尽的在线帮助,用户无须担心不能适应。CPAV 还具有可扩充性,用户只须按最新病毒资料输入 CPAV,便可扩充 CPAV 的检测病毒种数,1.0 版可检测 608 种病毒。

除主文件外,CPAV 还有几个辅助程序,包括 VSAFE、VWATCH 和 BOOTSDEF。VSAFE 与 VWATCH 是驻留程序,通过检查中断发现病毒,当病

毒入侵时,发出警告,并能用种种方式瞒骗和阻止病毒对系统进行破坏,连未知病毒也能防止。BOOTSAEF是一个专门对付引导区病毒的程序,初次运行它时,它会在磁盘中产生两个文件:BOOT.CPS和CMOS.CPS,以保留引导区和CMOS的信息,如发现引导区或CMOS遭改动(病毒感染或系统升级),它会发出警告,用户可选择更新或覆盖来恢复引导区。

笔者用了CPAV近一年,发觉有许多优点:如自动安装,操作简易和支持Windows等。还有几次靠它来除去一些病毒,例如旅游者(又名北京)病毒,虽然它不认识,但仍找到了遭感染改变的文件。

●但它仍存在一些问题,值得注意:

1. VWATCH使系统变慢,利用SCAN检查时,

SCAN会误以为有Invader(入侵者病毒)在内存活动。

2. 对于一些病毒太敏感,笔者曾对一硬盘用CPAV检查,它竟说有香港(Azusa)病毒,但诸如SCAN 3.1、SCAN 89、WATCHMAN等皆没法查出。

3. 并非所有可执行文件都可进行免疫,应事先做好备份及选择好非免疫对象。面对CPAV的执行文件,应首先进行免疫,防止病毒侵染它们,还应尽早对硬盘运行BOOTS SAFE,以防万一。

4. VSAFE已被某些病毒(诸如DIR2)避开,所以不可尽信:最重要的是自己小心。

总的说来,CPAV是优良的消毒软件。1992年推出的1.2版已可侦测过千种病毒。最新版1.5也已推出,让我们拭目以待。

用“锁匙软盘”保护硬盘

石家庄铁道学院(050043) 崔来堂

一、问题

如何保护硬盘数据,为大家所关注。目前多提倡“密码(口令)判别法”,思路是:在硬盘的适当部位,写入一段密码(口令)判别程序;自举时,控制先执行这段程序,提示用户键入密码,正确则正常自举,否则拒绝自举。具体方法大致分为两类:一是把密码程序插到主引导扇中主引导记录与分区表之间的空闲区域;二是把密码程序写到DOS分区引导扇的后部,覆盖掉这里的错误提示信息。但是这些方法都需向硬盘的要害部位手工插写较多内容,因DOS版本不同,插写的具体部位还要有所差异,实现起来比较困难,且容易出错,造成难以预料的后果;由于引导扇区的开放性,所加的密码或口令也容易被他人分析出来。

笔者今提出一种“锁匙软盘”法,同样可以达到保护硬盘数据的目的,但比较安全和简单易行。

二、方法

该法是由专用的锁匙软盘对硬盘进行加锁和解锁。

1. 生成锁匙软盘

(1)由C:\FORMAT A:/S格式化一张软盘,含有与硬盘版本相同的操作系统,再向该盘拷入DEBUG.COM文件。

(2)用DEBUG的A命令分别编写三个程序小段(见三),用N命令命名为PROF1.COM、PROF2.COM和PROF3.COM,用R和W命令写到这张软盘上。

(3)运行PROF1,读出硬盘主引导扇,分别写到该软盘的两个扇区S1和S2。这两个扇区要选在盘空间的偏后部位,以免破坏有用数据。

(4)将PROF1中给定的S1物理扇区参数,利用下式换算为逻辑扇区(针对360K软盘):

逻辑扇号 = 磁道号 * 18 + 磁头号 * 9 + 扇区号 - 1

并将结果转换为十六进制数,作为DEBUG的L命令的参数,读出S1扇区的内容。

(5)硬盘主引导扇的后部是4个分区表,最后是盘的自举标志“55AA”。每个分区表16个字节,具体内容是:

Boot -ind	Hs	Ss	Cs	SYS -ind	He	Se	Ce	Rel sect	# of sects
--------------	----	----	----	-------------	----	----	----	-------------	---------------

字节位移 0 1 2 3 4 5 6 7 8 12 15
其中:

- ① BOOT-ind 自举分区标志字节。其值为80H时表示该分区可自举。
- ② SYS-ind 系统特征字节。其值取:
 - 01 表示该分区为DOS分区,且FAT表的簇项长度为12bit;
 - 04 表示该分区为DOS分区,且FAT表的簇项长度为16bit。
- ③ Hs、Ss、Cs和He、Se、Ce分别为该分区起始和终止物理地址的磁头号、扇号、柱面号。
- ④ Rel sect 该分区相对起始扇号。
- ⑤ # of sects 该分区实用扇数。

用DEBUG的E命令改写DOS分区表中的个别信息。经多次实验知,只要进行下述内容中的一项,就可使硬盘变为“无效盘”(既不能从硬盘启动,用软盘启动后也不认硬盘,或硬盘不能正常工作):将SYS-ind改为非01、04,增大Hs或Ss或Cs,增大Rel sect,改变自举标志55AA。改后用W命令写回S1扇。改写的內容可过一段时间变更一次,以便增强保密性。

至此,就生成了一张针对该硬盘的锁匙软盘,贴上写保护,妥为保存。

2. 应用锁匙软盘

欲对硬盘加锁,只需运行锁匙软盘中的PROF2,

启动 DOS 的技巧

郑州市技工学校(450006) 邵爱柏

把改写过的 S1 扇内容写入硬盘主引导扇,硬盘就会暂时失效。主人使用时,先用锁匙软盘启动,并运行 PROF3,把 S2 扇中原硬盘主引导扇内容恢复回去,使硬盘解锁,再从硬盘重新启动,即可使用。

需要注意的是,硬盘处于加锁状态时,不要在软盘运行 PROF1。

该方法把对硬盘引导扇的改造,变成了在软盘上的简单操作,只要了解 DEBUG 和磁盘中断 INT 13H 基本用法的用户都可进行;被修改的个别字节也不易被辨认出来;又在锁匙软盘中可靠地保留了硬盘主引导扇正确内容的副本。因此,这是保护硬盘数据的一种简单、实用的方法。

三、程序

用 DEBUG 反汇编出来的程序清单如下(略去了地址编码和指令代码部分):

PROF1.COM:

```
MOV AX,0201
MOV BX,0200
MOV CX,0001
MOV DX,0080
INT 13
MOV AX,0301
MOV CX,2001
MOV DX,0100
INT 13
MOV AX,0301
MOV CX,2002
INT 13
INT 20
```

PROF2.COM:

```
MOV AX,0201
MOV BX,0200
MOV CX,2001
MOV DX,0100
INT 13
MOV AX,0301
MOV CX,0001
MOV DX,0080
INT 13
INT 20
```

PROF3.COM:

```
MOV AX,0201
MOV BX,0200
MOV CX,2002
MOV DX,0100
INT 13
MOV AX,0301
MOV CX,0001
MOV DX,0080
INT 13
INT 20
```

CEC—I 中华学习机在引导磁盘 DOS 过程中,如果能显示汉字,字符、奏乐,一定会给您的软件增色不少。同时,这种功能的实现,既不占用磁盘有用空间,又不占用内存空间,DOS 启动完毕,此功能即消失。

实现方法,和任晓方等先生所用的方法一样也是利用 DOS 缓冲区 \$BB00—\$BC55,(对应于磁盘 DOS“空区”0 轨 5 扇区、0 轨 6 扇区)写入相应的机器码及有关数据(乐谱信息,字符及汉字的三字节机内码)。

首先,运用《磁盘医生》等能读写磁盘扇区信息的工具软件,将 0 轨 0 扇区第 \$4A、\$4B、\$4C 三字节内容 \$6C,\$FD,\$08 改为 \$4C,\$00,\$BB,写入磁盘(见表①)

其次,再将下面一段机器码及信息资料写入磁盘的 0 轨 5 扇区(对应为 \$BB00—\$BBFF)和 0 轨 6 扇区(\$BC00—\$BC3F),。(见表②)

程序说明:① \$BB00—\$BB04:第一次进入中文状态时不显示中文商标

② \$BB05—\$BB0F:在高分辨第二页显示状态下打上黄色底色

③ \$BB10—\$BB1A:中文状态初始化,设置字符、文字在屏幕上显示位置

④ \$BB1B—\$BB27:计数、到字符,汉字存放区逐个取其机内码(汉字为 7F 开头的三字节内码)至显示完毕

⑤ \$BB0A:颜色代码(\$00—\$FF)

\$BB14:行号;\$BB18:列号

\$BB1E、1F:字符、汉字内码存放区地址、低位、高位字节

\$BB25:字符、汉字内码字节数

⑥ \$BB28—\$BB4C:奏乐程序

⑦ \$BB4D—\$BB5F:恢复 DOS 自举功能

⑧ \$BB60—\$BB7F:音乐乐谱存放区,末尾以 00 结束

⑨ \$BB80—\$BC3F:字符、汉字机内码存放区

⑩ 字符、汉字机内码查寻办法:清内存,打入 10 PRINT “字符、汉字”,进入监控,从 806 以后开始观察记录,以便存入 \$BB80 适当区域(写入盘内)

表①用《磁盘医生》等工具将 0 轨 0 扇区划线部位的三字节内容改为 4C 00 BB

```
00:01A5 27C9 09D0 18A5 2B4A 4A4A 4A09 C085
10:3FA9 5C85 3E18 ADFE 086D FF08 8DFE 08AE
20:FF08 3015 BD4D 0885 3DCE FF08 ADFE 0885
30:27CE FE08 A62B 6C3E 00EE FE08 EEF E0820
```


40:89FE 2093 FE20 2FFB A62B 6CFD 0800 0D0B
50:0907 0503 010E 0C0A 0806 0402 0F00 2064
.....

表② 将中间的机器码写入 0 轨 5 扇、6 扇区

* BB00:A9 03 8D 7B 04 A9 40 85
* BB08:E6 A9 D5 85 1C 20 F6 F3
* BB10:20 00 C3 A9 02 85 25 A9
* BB18:00 85 24 A2 00 BD 80 BB
* BB20:20 2B C3 E8 E0 C0 D0 F5
* BB28:A0 01 98 48 A2 00 BD 60
* BB30:BB F0 13 85 EC E8 BD 60
* BB38:BB 85 ED 8A 48 20 F9 D0
* BB40:68 AA E8 4C 2E BB 68 A8
* BB48:88 D0 DF A2 60 A9 6C 8D
* BB50:4A B6 A9 FD 8D 4B B6 A9
* BB58:08 8D 4C B6 4C 00 B7 60
* BB60:80 6E 80 46 72 46 AB A0
* BB68:C0 6E C0 46 E4 46 AB A0
* BB70:00 00 00 00 00 00 00 00
* BB78:00 00 00 00 00 00 00 00
* BB80:25 25 25 25 25 25 25 25
* BB88:25 25 25 25 25 25 25 25
* BB90:25 25 25 25 25 25 25 25
* BB98:25 25 25 25 25 25 25 25
* BBA0:25 25 25 20 20 20 20 7F
* BBA8:1D 6E 20 7F 55 4F 7F 39
* BBB0:27 7F 50 24 7F 4E 2E 7F
* BBB8:39 79 7F 4E 33 7F 4C 31
* BBC0:7F 44 4B 20 7F 1D 6E 20
* BBC8:20 20 20 20 20 20 25 20
* BBD0:20 20 20 20 20 20 20 7F
* BBD8:1F 42 7F 1F 44 7F 1F 42
* BBE0:7F 1D 27 7F 1F 43 7F 1F
* BBE8:4E 7F 1F 52 20 20 20 20
* BBF0:20 20 20 20 20 20 20 25
* BBF8:20 20 20 20 20 20 20 20
* BC00:7F 2F 5F 7F 3B 2A 20 20
* BC08:20 20 7F 48 5A 7F 2E 2B
* BC10:7F 2E 57 20 20 20 20 20
* BC18:20 20 20 20 20 20 25 25
* BC20:25 25 25 25 25 25 25 25
* BC28:25 25 25 25 25 25 25 25
* BC30:25 25 25 25 25 25 25 25
* BC38:25 25 25 25 25 25 25 25

中华学习机 DOS 使用技巧

福建宁化师范学校(365400) 张本灵

(1) 设置缓冲区的大小

若你需要在引导 DOS 的同时就设置好缓冲区的个数,那么,只要修改一下 \$AAB1 的值就行了。比如说为 2,指的是可使用的缓冲区是 \$9853—\$9CF8 两个,这样你可以把 \$9600—\$9852 作它用。如果你打

算把这两个缓冲区设置在 \$9600—\$9AA5,而想用 \$9AA6—\$9CFE 来放机器语言子程序,那么,在改变 \$AAB1 为 2 的同时,还要将第一个文件的缓冲区首址(其实是文件名缓冲区的首址)设置为 \$9A80,即作如下改动:\$9D00:80 9A

(2) 设置“&”和“^ Y”的入口

假如你在 DOS 的空区内放置了扩充“&”和“^ Y”的程序,那么,你当然希望在引导 DOS 的同时就设置好“&”和“^ Y”的入口,DOS 中设置第三页向量是由 \$9E25—\$9E40 这一段程序完成的,具体的向量值在 \$9E51—\$9E80 中,其中 \$9E77、\$9E78 为“&”的入口,\$9E7A、\$9E7B 为“^ Y”入口,你只要将这四个单元修改为相应的处理程序的入口地址就行了。

作了以上修改后,若 INIT 一张磁盘,那么这张盘的 DOS 便自动设置好了上述的参数和地址。对于已经存有文件的盘片,可利用修改磁道扇区的工具软件来修改。

(3) 第 0 道,第 10、11 扇区的使用

我们知道,对于 DOS 主盘来说,第 0 道的 10、11 扇区存放的是一段移动程序,用主盘来引导 DOS 时,它负责将 DOS 操作系统程序从 \$1D00—\$3FFF 移到当前系统的最高 RAM 内存位置(CEC—I 上为 \$9D00—\$BFFF)。而对于从属盘来说,这两个扇区上没有放任何程序。但是,在用从属盘来引导 DOS 时,BOOT2 也会将这两个扇区的内容自动装入内存(在 CEC—I 上为 \$9B00—9CFE,是第一文件缓冲区的一部分)。我们平常使用的一般都是从属盘,因此,只要在这两个扇区中存放一些实用的机器语言程序(用读写磁盘程序很容易做到这一点,因为有 2 页的空间,足够放个较长的程序),那么,你一引导 DOS,这些程序便进入了 \$9B00—\$9CFE 中。只要你在后面的使用中同时打开的文件数不超过 2 个,这些程序便一直可以使用,你若要无条件使用这些程序,可参照(1)来进行。

(4) VTOC 表缓冲区中空区的使用

在 DOS3.3 中,\$B3BB—\$B4BA 为 VTOC 表缓冲区,从 VTOC 表的使用情况可以知道,在这一页的空间中,有 90 多个单元用户可以使用的,分为两部分:1) \$B3C2—\$B3EA,2) \$B47F—\$B4BA(若格式化到 40 磁道,则相应减少 16 个单元,为 \$B493—\$B4BA)。对这些空区的使用如果象平常使用空区一样,直接把程序写进这些单元,这是不够的,为什么呢?因为 DOS 一进行文件操作,便将第 17 道第 0 扇区的 VTOC 表读进该缓冲区因而冲掉数据,唯一的办法是,将数据写进 VTOC 表中,你可以用编辑磁道扇区的工具软件,将程序写至第 17 道第 0 扇区 VTOC 表中的 \$07—\$2F 和 \$C4—\$FF 中。这样,当你没有操作磁盘时,程序已在内存,而每当你进行磁盘操作时,这些程序便会又一次随着 VTOC 表进入缓冲区。你就能够放心地使用空区中的程序了。

学习机游戏软件“攻关秘诀”

北京和平街 13 区 9 楼 4 门 202 号
(100013) 王海涛

中华学习机的游戏软件非常丰富,而且软件设计得都十分精巧,充分利用学习机的 48K 内存,所以软件读起来就比较难懂。笔者对多个游戏进行了研究,发现了游戏软件改命、加命(延长游戏次数)的方法,现介绍给大家,与各位交流一下游戏软件的“攻关秘诀”。

一、打臭虫

英文名:BUG ATTACK

文件长度及首地址:7FD0、800

控制键:按“K”开始,用键盘控制,←、→和空格。

按游戏杆的按钮 0 开始用游戏杆操纵

加命方法:见后面,下面划横线的地方,改成后面的数字(CR)表示回车

二、大赛车

英文名:AUTOBAHN

控制键:按游戏杆按钮“0”控制速度,按键盘 1、2、3 键也是控制速度,P、K 键选择游戏杆或键盘。

文件首址及长度:800 和 5200

三、空投

英文名:STAR BLAZER

控制键:←、→、A、Z、空格

文件首址及长度:FD0、7200

四、大嘴巴

英文名:Puc'man

控制键:K 选择键盘,操纵 I、J、K、M 控制方向,P 选择游戏杆,D 演示,Q 控制声音开关,C 控制颜色,S 游戏开始。

文件首址及长度:800、8800

五、打太阳

英文名:PULSAR

控制键:←、→、空格

文件首址及长度:IEFD、7790

任意数的高精度除法运算程序

山东苍山县酒厂(277700) 张新莲

所有微机都能进行除法运算,但其处理数据的能力和运算精度却有天壤之别。如中华学习机能处理 38 位长的数据;运算精度有九位有效数字,而电视游戏学习两用机及其它使用整数 BASIC 语言的机型则只能处理 $-32768 \sim +32767$ 之间的数据,且商没有尾数部分。与其相比,前者的运算能力要大得多,运算精度也高得多。但它仍不是高精度的运算。

为求得高精度的结果,人们进行了探讨研究,大量文献都介绍了不同的高精度除法运算程序。这些程序,异曲同工,都可以求出小数点后任意位尾数。但是,也都存在一个共同的缺陷,就是只能高精度计算九位数字以内的除法运算(即小于 10 亿的数据)。对多于九位的数,虽然仍能求出任意位尾数,但这时参与运算的数据已使用科学记数法表示,本身已不精确,致使商失去了精确性。

能否对任意数进行高精度除法运算呢?本文介绍的程序可以解决这一问题。

前文介绍的任意数高精度乘法运算、加减运算程序都通过使用字符串装数据实现了高精度运算。万变不离其宗,除法的高精度运算仍然使用这一方法。

与前三种运算相比,除法运算要复杂得多。众所周

知,在除法运算中包含了减法、乘法、数的分解与合成、试商的判断等多种操作过程。因此,除法运算过程最为复杂,商的形式多种多样。这就要求程序对各种算式的变化做出正确的判断,确定正确的商的形式。我们知道在以往介绍的高精度除法程序中,商的整数部分可通过 $\text{INT}(X/Y)$ 一次确定,然后加小数点,对余数补“0”,依次求出任意位尾数。而使用字符串装数据,这一方法显然已不适用。因此,本文程序中使用了 A、B、C、D、E、G、Z 几个简单变量,通过它们之间的相互辅助和制约,巧妙地解决了商的各种形式的输出打印。其中,A、B、Z 分别为被除数、除数、余数的字符长度;C 为运算精度控制变量;D 记录本次运算中商是否添过“0”或小数点;E 记录余数前面是否为“0”,以便确定不够除时商是否添“0”;G 记录被除数是原始输入的,还是再生的(所谓再生,是指通过一次求商运算,余数加上被除数的剩余部分或补“0”,形成新的被除数的过程。这样的被除数称为再生被除数。),若是原始的,不够除时,商要打印“0.”;否则,只打印“.”。另外,使用字符串装数据为试商的判断带来了困难。为压缩程序量,本文采用了珠算除法中“小扒皮”(以减代除)的方法完成了除法运算,从而使运算程序短小精练。这样一来,虽然求

一位商要进行多次减法运算,但由于程序短小,运算速度并不减慢。

本程序已在 CEC— I 机上通过。也可在 LASER、FCS—90、游戏学习两用机等多种机型上运行。

程序说明:

1. 25~35 行为被除数为“0”时的处理程序。
 2. 40~50 行为除数为“0”时的处理语句。
 3. 55~65 行为两数负号的判别、去除及商的符号确定程序。
 4. 70~120 行的作用是去小数点,把两数均化为整数。
 5. 135~150 行为被除数的位数少于除数的位数时,对被除数补“0”和对商的处理程序。
 6. 155~160 行的作用是从被除数中取出与除数位数相等的一组数,形成子被除数。原理与笔算相同。
 7. 165~180 行是比较子被除数与除数两数绝对值大小的程序。
 8. 200~230 行的作用是当子被除数小于除数时,对子被除数增加一位或添“0”的处理以及对商的处理。
 9. 240~275 行是除法运算程序。这里使用了变量 T 记录商。其中 265 行是余数的记录程序。C(B)数组中始终记录着上一次减法运算后的余数,以保证余数转移的正确性。
 10. 285~330 行是对余数的不同情况进行处理程序
 11. 335~345 行是被除数的再生程序,通过该部分处理后的被除数称为再生被除数。
 12. 350、355 行是精度控制程序。当达到要求精度时,则结束运算。
- 运行程序,依次输入被除数、除数和精确度要求,即可打印结果。
- ```
10 HOME
15 CLEAR:U=0;V=0;T=0;C=0;D=0;E=0;G=0
20 INPUT "X/Y.....N= ";X$,Y$,N;PRINT X$"/"
 Y$"$";A=LEN(X$);B=LEN(Y$)
25 FOR I=1 TO A:XX=VAL(MID$(X$,I,1))
30 IF XX>0 THEN 40
35 NEXT;PRINT "0";END
40 FOR I=1 TO B:YY=VAL(MID$(Y$,I,1))
45 IF YY>0 THEN 55
50 NEXT;PRINT;PRINT"ERROR! INPUT AGAIN!";
 GOTO 15
55 IF LEFT$(X$,1)="-" THEN A=A-1;X$=
 RIGHT$(X$,A);U=1
60 IF LEFT$(Y$,1)="-" THEN B=B-1;Y$=
 RIGHT$(Y$,B);V=1
65 IF U<>V THEN PRINT "-";
70 FOR I=1 TO A;IF MID$(X$,I,1)="-" THEN P=
 A-I;GOTO 80
```

```
75 NEXT;GOTO 90
80 A=A-1;X$=LEFT$(X$,A-P)+RIGHT$(X$,P)
85 IF LEFT$(X$,1)="-" THEN A=A-1;X$=
 RIGHT$(X$,A);GOTO 85
90 FOR I=1 TO B;IF MID$(Y$,I,1)="-" THEN Q=
 B-I;GOTO 100
95 NEXT;GOTO 110
100 B=B-1;Y$=LEFT$(Y$,B-Q)+RIGHT$(Y$,Q)
105 IF LEFT$(Y$,1)="-" THEN B=B-1;Y$=
 RIGHT$(Y$,B);GOTO 105
110 IF P=Q THEN 125
115 IF P>Q THEN FOR I=1 TO P-Q;Y$=Y$+"0";
 B=B+1;NEXT
120 IF Q>P THEN FOR I=1 TO Q-P;X$=X$+"0";
 A=A+1;NEXT
125 DIM A(B),B(B),C(B);FOR I=1 TO B;B(I)=VAL
 (MID$(Y$,I,1));NEXT
130 A=LEN(X$)
135 IF A<B AND G=0 AND C=0 AND D=0 THEN
 PRINT"0.";X$=X$+"0";A=A+1;C=1;G=1;
 D=1
140 IF A<B AND C=0 THEN PRINT ". ";X$=X$
 +"0";A=A+1;D=1;C=1
145 IF A<B AND D=0 THEN X$=X$+"0";A=A+
 1;D=1
150 IF A<B AND D=1 THEN PRINT "0";X$=X$
 +"0";C=C+1;A=A+1;GOTO 150
155 X1$=LEFT$(X$,B);IF A=B THEN X2$="?";
 GOTO 165
160 X2$=RIGHT$(X$,A-B)+"?"
165 FOR I=1 TO B;X1=VAL(MID$(X1$,I,1));Y1=
 VAL(MID$(Y$,I,1))
170 IF X1>Y1 THEN 185
175 IF X1<Y1 THEN 195
180 NEXT
185 FOR I=1 TO B;A(I)=VAL(MID$(X1$,I,1));
 NEXT
190 A(0)=0;GOTO 240
195 IF A>B AND E=1 THEN PRINT "0";GOTO 225
200 IF A=B AND G=0 AND C=0 AND D=0 THEN
 PRINT "0.";X$=X$+"0";C=1;A=A+1
205 IF A=B AND E=1 AND C=0 THEN PRINT "0.";
 X$=X$+"0";C=1;A=A+1
210 IF A=B AND C=0 THEN PRINT ". ";X$=X$
 +"0";A=A+1;C=1
215 IF A=B AND D=0 THEN X$=X$+"0";A=A+
 1
220 IF A=B AND D=1 THEN PRINT "0";X$=X$
 +"0";A=A+1;C=C+1
```

```

225 A=LEN(X$);X1$=LEFT$(X$,B+1);IF A=B
 +1 THEN X2$="?";GOTO 235
230 X2$=RIGHT$(X$,A-B-1)+"?"
235 FOR I=0 TO B;A(I)=VAL(MID$(X1$,I+1,1));
 NEXT
240 X=B
245 FOR I=B TO 1 STEP-1
250 IF A(0)*10+A(1)<B(1) THEN 275
255 IF A(X)<B(I) THEN A(X)=A(X)+10;A(X-1)=
 A(X-1)-1
260 A(X)=A(X)-B(I);X=X-1;NEXT
265 IF X=0 THEN FOR J=1 TO B;C(J)=A(J);NEXT
270 T=T+1;GOTO 240
275 PRINT T;:E=0;T=0;Z$="0"
280 FOR I=1 TO B;Z$=Z$+STR$(C(I));NEXT
285 IF X2$="?" THEN X$=Z$;GOTO 340
290 Z=LEN(Z$);X2=LEN(X2$);Z=Z-1;Z$=
 RIGHT$(Z$,Z);X2=X2-1;X2$=LEFT$(X2
 $,X2)
295 IF X2>0 THEN Z$=Z$+LEFT$(X2$,1);Z=Z
 +1;X2=X2-1
300 IF X2>0 THEN X2$=RIGHT$(X2$,X2)
305 IF LEFT$(Z$,1)="0" AND Z>1 THEN Z$=
 RIGHT$(Z$,Z-1);Z=Z-1;E=1;GOTO 305
310 IF Z=1 AND Z$="0" AND X2=0 THEN PRINT"
 0";:END

```

```

315 IF Z=1 AND Z$="0" AND X2>0 THEN PRINT"."
 0";:GOTO 295
320 IF X2=0 AND Z<B THEN PRINT"0";:X$=Z$:
 GOTO 360
325 IF X2=0 AND Z>=B THEN X$=Z$;GOTO 360
330 IF Z<B THEN PRINT"0";:GOTO 295
335 X$=Z$+X2$
340 A=LEN(X$);IF A=1 AND X$="0" THEN END
345 IF LEFT$(X$,1)="0" THEN X$=RIGHT$(X
 $,A-1);GOTO 340
350 IF C>0 THEN C=C+1
355 IF C>N THEN END
360 D=0;G=1;GOTO 130

```

本刊登在今年第二期上的“任意位小数，整数的连续加减精确运算程序”不能正常运行，经核对，发现程序清单中有四处错误，其中有一处错误的影响较大，在此向读者致谦并更正于下：

1. 145 行应为 IF R¥(K)="." THEN R¥(K) = "0."
2. 225 行应为 IF I=P+1 THEN I=I-1;
3. 305 行应为 A=VAL(MID¥(X¥,I,1));
4. 305 行应为 FOR I=1 TO N;.....

## ProDOS 系统内部结构剖析(续)

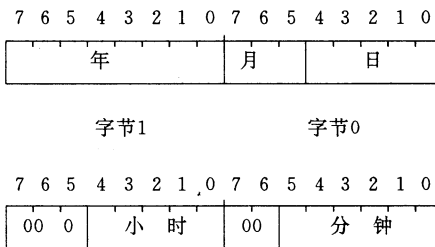
北京铁路局卫生防疫站(100038) 廖凯

### 四、磁盘结构

在一卷上块的总体结构和标准文件的三种类型的整个结构已在前面介绍过，下面对一些具体字段进行介绍。

#### 1. 建立、修改日期及时间(4 字节)

每个文件和目录的建立日期和时间与最近一次修改的日期和时间被存储在二个四字节内，格式如下：



对于年、月、日、小时和分钟的值均为二进制整数。

#### 2. 存储类型(4 位)

存储类型在一个项目的首字节的高四位，它定义标题(如果此项目是标题)的类型或被该项目描述的文件类型。

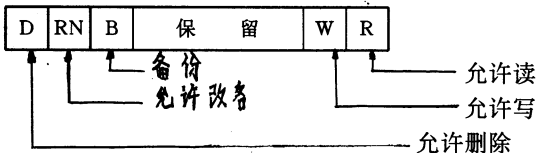
- \$0 表示一个无效文件项目
- \$1 表示一个树苗型文件项目 (EOF ≤ 256 字节)
- \$2 表示一个幼树型文件项目 (256 (EOF ≤ 128K 字节)
- \$3 表示一个树型文件项目 (128K (EOF ≤ 16M)
- \$D 表示一个子目录文件项目
- \$E 表示一个子目录标题
- \$F 表示一个卷目象标题

文件类型(1字节):

- \$ 00 无类型文件
  - \$ 04 ASCII 文本文件
  - \$ 06 一般二进制文件
  - \$ 0F 目录文件
  - \$ C0—EF ProDOS 保留
  - \$ F0 新增命令文件
  - \$ F1—F8 用户定义文件 1—8
  - \$ F9 ProDOS 保留
  - \$ FC Applesoft 程序文件
  - \$ FD Applesoft 变量文件
  - \$ FE 浮动代码文件(EDASM)
  - \$ FF 系统文件
- 其余为 Apple II 的 SOS 系统专用

存取标记(1字节):

存取标记决定文件是否可以被读取、写入、删除或改名,它也可以用来指示在文件最近一次修改后是否建立了文件的备份。



位元为 1,表示允许操作;位元为 0,表示不允许操作,保留位总是 0。每当文件被改变(建立、更名、写后关闭或 SET-FILE-INFO 操作后,ProDOS 置备份位(位 5)为 1。此位应在文件被复制备份后设为 0。

项目字段:

- 名字长度 文件名的长度(\$ 1—\$ F)
- 文件名 首字符为字母,\$ 1—\$ F 个 ASCII 字符,其余字符可为字母数字或句点。
- 关键块指针 文件的关键块的块地址
- 使用块数 文件所占用的块数
- EOF 文件结尾的字节数(\$ 0—\$ FFFFFFF)
- 版号、最小版号 对于 ProDOS 1.1.1 为 0
- 项目长度 对于 ProDOS 1.1.1 为 \$ 27
- 每块项目数 对于 ProDOS 1.1.1 为 \$ 0D
- 辅助类型 由系统程序定义
- 现有文件数 目录内文件总数
- 位图指针 位图的块地址
- 总块数 卷上块总数
- 根指针 包含项目的块地址
- 根项目号 在那个块内的项目号
- 根项目长度 对于 ProDOS 1.1.1 为 \$ 27
- 标题指针 项目的目录的关键块的块地址

### 五、DOS3.3 磁盘结构

DOS 3.3 和 ProDOS 140K 软盘使用相同的 16 扇区格式。因此,ProDOS READ-BLOCK 和 WRITE-

BLOCK 调用也能存取 DOS 3.3 磁盘。

在使用 READ-BLOCK 和 WRITE-BLOCK 时,你在磁盘上指定一个 512 字节块,在使用 RWTS(DOS 3.3 对应的 READ-BLOCK 和 WRITE-BLOCK)时,你指定一个 256 字节块的数据的磁道和扇区,参见 DOS 手册的说明。要使用 READ-BLOCK 和 WRITE-BLOCK 存取 DOS 3.3 磁盘时,你必须知道 512 字节对应的磁道和扇区,磁道和扇区与块号的换算公式如下:

$$\text{块} = (8 \times \text{磁道}) + \text{扇区位移量}$$

扇区号 0 1 2 3 4 5 6 7 8 9 A B C D E F

扇区位移量:0 7 6 6 5 5 4 4 3 3 2 2 1 1 0 7

半块:1 1 2 1 2 1 2 1 2 1 2 1 2 1 2 2

扇区所在的半块由“半块”行决定(1 是首半块,2 是后半块)。

## 第四章 存储区的使用

本章介绍 MLI 使用存储区的方法,描述系统程序使用内存的情况以及系统程序怎样管理空闲存储区。

### 一、载入顺序

当你由 ProDOS 启动 APPLE II 时,便开始装载 MLI 和磁盘上的引导程序,一个初级装载程序存在磁盘驱动接口卡上的只读存储器 ROM 内;引导程序大部分存在每个 ProDOS 格式磁盘的 0 和 1 块上。

当你启动计算机,用一个 PR# 或 IN# 命令来启动一个磁盘驱动器后,在 ROM 内的程序读取磁盘的 0 和 1 块的引导程序,并放置到内存 \$ 800 处,然后执行它。这引导程序在启动盘的卷目录上查找名为 PRODOS 和类型为 \$ FF 的文件(包含 MLI),放置到内存 \$ 2000 处并执行。MLI 确定计算机的存储器的大小并移动它自身到最高端。接着它确定在槽口上连接的设备,建立系统整体页面。MLI 然后寻找启动盘卷目录中第一个名为 XXX.SYSTEM 和类型为 \$ FF 的文件,将它放置到内存 \$ 2000 处并执行。

若没找到 PRODOS 或系统文件,则系统通知用户装载 ProDOS 不成功。

MLI 是常驻内存的,一旦它在内存中,它就不再移动,也不需要任何附加的磁盘存取。

### 二、卷检索次序

当一个程序或用户需要存取一个 ProDOS 还未存取的卷时,它必须搜寻所有存在的卷。要搜寻的第一个卷总是启动的卷,其次从槽口 7 开始降序搜索,驱动器 1 在驱动器 2 之前搜索。

例如,若在槽口 6 有两个驱动器,在槽口 5 有两个驱动器和在槽口 7 有一个 ProFile 硬盘,检索次序是:

槽口 6,驱动器 1

槽口 6,驱动器 2

槽口 7



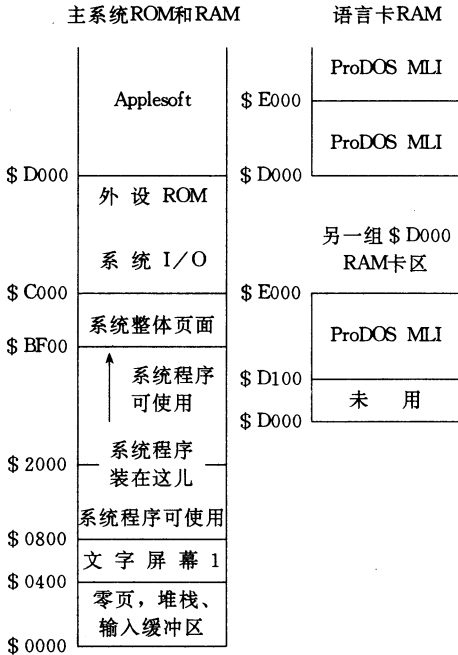
槽口 5, 驱动器 1

槽口 5, 驱动器 2

上例中启动卷在槽口 6 内。

### 三、存储区分布图

ProDOS 有一种基本存储区配置: 一个对于 AP-  
PLE IIe 的配置或一个对于 64K APPLE II PLUS 的  
配置, 分布图如下:



一个象 \$ 8F00(36608)字节的系统程序可被装入  
到一个 64K 系统中, 在一个 64K 系统上运行的系统程  
序的可使用空间为 \$ B700(46848)字节。

#### 1. 零页

ProDOS MLI(机器语言接口)使用零页 \$ 40—  
\$ 4E 单元, 但在它完成一个调用之前恢复其值。磁盘

驱动程序经 MLI 调用, 使用 \$ 3A—\$ 3F 单元, 这些单  
元的值不被恢复。

#### 2. 系统整体页面

内存 \$ BF 页, 即 \$ BF00—\$ BFFF, 包含系统的  
整体(全局)变量。内存的这部分是专用的, 因为无论在  
什么系统上启动 ProDOS, 这整体页面总是在相同  
的位置上, 由于这, 它在系统程序和操作系统之间起着通  
信连接作用。MLI 在这些单元内放置对系统程序有用的  
所有信息。

#### 3. 系统位图

ProDOS 用一个简单的形式管理存储器, 允许它  
保护它自身及它的数据以免被重写, 它用系统整体页  
面的 24 个字节表示低 48K APPLE II 的随机存取存  
储器: 每一位对应于低 48K 的 APPLE 机 RAM 的每  
256 字节页, 这 24 个字节被称为系统位图。

在 MLI 被装入时, 所有它的程序及数据区在系统  
位图中被记为已用; 也就是说, 设置对应于使用的页的  
位。如果可能, 一个系统程序不应使用已用的存储页,  
如果不可能, 系统程序会关闭所有文件和清除位图  
——从页 \$ 00 到页 \$ BE, 如果在关闭时有错误发生,  
程序会要求用户重新启动系统。

一个系统程序在使用 MLI 时, 只有三种调用对位  
图起作用: OPEN, CLOSE 和 SET\_BUF。在系统程序  
打开一个文件时, 它必须指定一个 1024 字节的文件缓  
冲区的起始地址。只要文件被打开, 这缓冲区是系统  
的一部分, 并在位图内被标记, 当文件被关闭时, 这缓  
冲区被释放, 其位被清除。

一般来说, 系统程序要求使用内存页是连续的。这  
样就可以把尽可能多的连续的内存空间留给数据。假  
设一个系统程序打开几个文件, 而后关闭第一个打开  
的文件, 在大多数情况下, 这会出现一个空白 1K 区  
域, GET\_BUF 和 SET\_BUF 调用可以用于寻找这样  
的空白区, 并移动另一个文件的缓冲区到这空间。

(未完待续)

## FORTH 语言讲座

# 第 6 讲 程序控制结构(上)

北京 FORTH 应用研究会(100034) 丁志伟

在第一讲中曾提到, FORTH 是天然的结构化语  
言, 其程序控制结构是比较完善的。所谓程序控制结  
构, 是指诸如分支、循环这类控制程序走向的机制, 它  
们在各单词的定义中随处可见。举个例子, PC/  
FORTH 2.0 版本中有一个打印屏幕文件的词

SHOW, 请看其定义:

```

SHOW DEPTH 2 < 1 ? ERROR
? DECIMAL PRINTER 1+ SWAP
DO I TRIAD ? TERMINAL
IF KEY DROP LEAVE

```

+LOOP CONSOLE ;

这个定义中包含了分支判断结构和循环结构,其中?ERROR、?TERMINAL 等词的构成也包含了控制结构。

在介绍各种结构之前,先来讨论一下在大多数高级语言中都有 GOTO 语句。程序一般是按顺序执行的,由于各种原因,有时程序可能需要直接转到其他位置去执行,这就需要 GOTO 语句。它给编程带来了灵活性,在某些语言(如 BASIC、FORTRAN)中,如果不用这个语句,有时会很非常不便。但使用它的结果,令程序走向难以把握,常常会造成程序错误,这就象车辆在缺乏交通法规约束的道路上行驶,方便中隐含着危险。有鉴于此,在 60 年代中后期,软件界有人主张取消 GOTO 语句,从而引发了一场论战,对这个语句的利弊展开了激烈的争论。现在普遍认为,这个语句虽然能在某些情况下提高程序灵活性和效率,但因为它容易带来副作用,所以应该限制它的使用范围。

一种语言要想限制 GOTO 语句的使用。就要有足够丰富的控制结构才行。60 年代末,被称为结构化语言的 PASCAL 应运而生了,但它仍保留了 GOTO 语句。

与 PASCAL 同一时期发明的 FORTH 语言不存在上述问题。笔者接触到的十几种版本,都没有 GOTO 这个词,从原理上,FORTH 语言可以容纳 GOTO,但并无此必要。也很少有 FORTH 编程者提出需要使用这个语句。值得一提的是,FORTH 中的汇编语言也可以不使用标号和不直接使用 JMP 指令,而是使用同 FORTH 中结构一样的方式来编程,或者说,FORTH 中使用的是结构化汇编。比如 D1.0 版本的汇编语言就不使用标号,有人说,FORTH 是天然的结构化语言,是很有道理的。

下面就来看看 FORTH 中都有什么结构。

- ① (f) IF ... THEN
- ② (f) IF ... ELSE ... THEN
- ③ ( n<sub>1</sub> n<sub>2</sub>) DO ... LOOP
- ④ ( n<sub>1</sub> n<sub>2</sub>) DO ... (n<sub>3</sub>) +LOOP
- ⑤ BEGIN ... AGAIN
- ⑥ BEGIN ... (f) UNTIL  
或 BEGIN ... (f) END
- ⑦ BEGIN ... (f) WHILE ... REPEAT
- ⑧ n CASE
  - (n<sub>1</sub>) OF ... ENDOF
  - (n<sub>2</sub>) OF ... ENDOF
  - (n<sub>3</sub>) OF ... ENDOF
  - .....

在这些结构中:

- ①和②是分枝结构。
- ③也是一种分支结构,但稍复杂,是多路分支结构。又称为分情况结构,多路开关。
- ④和⑤是固定循环,也叫限定循环。
- ⑥和⑦是不定循环。
- ⑧是无限循环,也叫死循环。

在某些版本中,可能会有其他结构,如? DO... LOOP、FOR...NEXT 等。

一般情况下,这几种结构已经够用。如果有必要,用户可以自己定义出新的结构,笔者接触到的第一个 FORTH 版本,不存在⑤、⑦、⑧这三种结构,有时不太方便,于是笔者将其定义了出来,并不太难,其中的 CASE 结构,笔者还设计出两种方案。关于③、④两种结构,其循环范围在某些场合下可能会嫌小,有人为此设计出 2DO ... 2LOOP,能把循环范围扩大几个数量级。可见 FORTH 的扩充能力是很强的。

前边列出这几种结构时,出现的(f),它代表标志,在介绍各种结构之前,这里先介绍一下标志的概念,同时介绍各种比较词和逻辑运算符。

所谓标志,是供某些结构判别程序走向的数据,也是放在堆栈中,它有两种值,真和假。有些版本中,真假又会用 true 和 false 来表示。栈中的数,做为标志时,0 代表假,非 0 的数都被认为是真。

标志一般是由比较操作得到的。它们是一组词。常用的有:

- > ( n<sub>2</sub> -- f)大于比较操作
- < ( n<sub>2</sub> -- f)小于比较操作
- = ( n<sub>2</sub> -- f)等于比较操作
- >=( n<sub>2</sub> -- f)大于等于比较操作
- <=( n n<sub>2</sub> -- f)小于等于比较操作
- <> ( n n<sub>2</sub> -- f)不等于比较操作

以大于号“>”为例,它将栈中两数的做比较,得出标志 f,在 n<sub>1</sub>>n<sub>2</sub> 时,标志为真,如果 n<sub>1</sub>≤n<sub>2</sub>,标志则为假。对于 D1.0 版本,比较运算结果为真时,是在栈中放一个 1。请看:

```

3 5 > .✓
显示 1
5 3 > .✓
显示 0
5 5 > .✓
显示 0

```

如果使用 PC/FORTH 版本,比较运算为真时,栈中不是 1,而是-1,这是因为这两个版本遵循不同标准的结果,对于大多数情况,它们的效果都是一样的。

在算术等运算时,经常要用堆栈中数与 0 做比

较。比如 a1 是一个变量，

```
a1 @ 0 >
```

这是将 a1 变量中的值与 0 做比较，看它是否大于 0。为了方便，多数版本都定义出一组与 0 比较的词。比如

```
0>(n ... f)与 0 做大于比较
```

与前边 >、= 这些比较操作不同，它只需要栈中有一个数。这样的词可以用汇编定义，也可以用冒号定义，

```
:0> 0 >;
```

相应地，还有 0=、0<、0>=、0<=、0<> 这几个词。

有时，把栈中数做为无符号数处理时，则要用无符号比较词，如 U>、U= 等。

PC/FORTH 中还使用双精度数，相应地也使用双精度比较词如 D=、D>、D< 等。

对于标志，还可以进行逻辑运算，它们分别是

```
AND (n1 n2 -- n3)与运算、逻辑乘
```

```
OR (n1 n2 -- n3)或运算、逻辑加
```

```
XOR (n1 n2 -- n3)异或运算、按位加
```

```
NOT (n -- n)对栈中数求反
```

对于这几个逻辑运算的词，需要有一些数字逻辑运算基础知识才能理解，这里只简单介绍一下。

比如 AND 是与运算，又叫逻辑乘，如

```
1 1 AND .
```

显示 1

```
1 0 AND .
```

显示 0

这些逻辑是按位运算，即分别对数据的每一位进行运算。或者说，逻辑运算不止用于标志。

如果要取出一个数的后 3 位二进制数，比如这个数是 15，就可以用：(10 进制下)

```
15 7 AND .
```

显示 7

限于篇幅，就不多介绍了，请读者参考有关资料。

下面就介绍各种结构。

第①种即 IF ... THEN 结构。先看看这个结构的一个应用实例。

PC/FORTH 中有一个 MAX 的词，能求出两数中较大的数。D1.0 版本中不存在这个词，可以定义出一个，它使用了 IF ... THEN 结构。

```
: MAX (n1 n2 ... nmax)
```

```
OVER OVER <
```

```
IF SWAP THEN DROP ;
```

请看这个词的详细注释。

: MAX ( n1 n2) 定义词名，执行前栈中要放好两个数

OVER OVER ( n1 n2 -- n1 n2 n1 n2) 复制 n1 和 n2，以供比较操作之用。

< ( n1 n2 -- n1 n2 f) 得出 n1 是否小于 n2 的标志，以决定是否执行 SWAP

IF (n1 n2 f -- n1 n2) IF 会“吃掉”标志

SWAP (n1 n2 -- n2 n1) 如果 f 为真，则交换

THEN 分枝结构结束

DROP ; ( n1 n2 -- n1) 丢掉栈顶较小的数 或 ( n2 n1 -- n2)

MAX 先把栈顶两数复制，之后进行比较，如果 n1 小于 n2，则执行 SWAP，这样在执行 DROP 之前，栈顶数总为较小的数，DROP 丢掉栈顶数之后，留下来的就是较大的数。

对于学习 FORTH 语言，各类结构是重点之一。下面从不同角度来看看 IF...THEN 结构。

这个结构，使用了英语词，翻译成汉语，是“如果...就...”的意思。在其他高级语言中，正是这么用的。但在 FORTH 中，因为使用逆波兰，含义有些差别。请看 FORTH 中这个结构与汉语含义的对应关系：

IF SWAP THEN DROP 如果(栈中标志)为真则执行 SWAP 然后 DROP

在笔者见到的一个汉化版本中，“IF”这个词被译为“若就”。

这里的 IF 和 THEN 又可看成一对语句括号，是一组操作的分界符号，用来控制程序运行方向，除了 IF 要“吃掉”栈标志外，不产生其他影响。请看其一般格式，设 W 为一组单词。

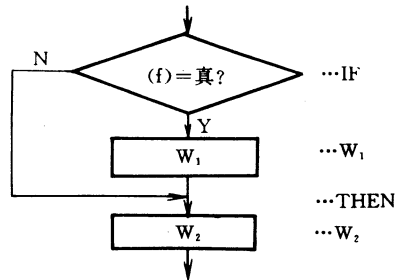
```
(f) IF W1 THEN (W2)
```

如果 f 为真，则执行 W1，继而执行 W2。

如果 f 为假，则不执行 W1，继而执行 W2。

这里的 W2，与这个结构没有关系，提到它，只是为了更清楚地说明执行顺序。执行 IF 时，会“吃掉”栈中的标志。

这个结构也可以用框图说明





这样的表示方法一目了然,就不多解释了。

如果读者懂得汇编语言,还可以发现,FORTH 中 IF...THEN 结构,用法与汇编中条件转移相似。这正是 FORTH 中结构化汇编的基础。请看:

汇编 与 FORTH 对应关系

```
MOV R1,n1 (--n1)
MOV R2,n2 (n1--n1n2)
CMP R1,R2 =
JZ LABEL IF
... ...
```

TABLE ... THEN

```
再举一例,先定义一个 TEST
: TEST CR ."This flag is "
 IF ." not "THEN
 ." false " CR ;
```

定义之后,

0 TEST (0 是供 IF 用的标志)

会显示 This flag is false  
(这个标志为假)

1 TEST

显示 This flag is not false  
(这个标志不为假)

接下来看第②种结构。

(f) IF W<sub>1</sub> ELSE W<sub>2</sub> THEN (W<sub>3</sub>)

这个结构与①相通,又有区别。如果对第①种有了充分理解,这个结构也就不难理解了。

当 f 为真时,则执行 W<sub>1</sub>,跳过 W<sub>2</sub>,然后执行 W<sub>3</sub>。

当 f 为假时,则跳过 W<sub>1</sub>,执行 W<sub>2</sub>,然后执行 W<sub>3</sub>。

请看例子。

```
: TEST CR ." This flag is"
 IF ." true
 ELSE ." false"
 THEN CR ;
```

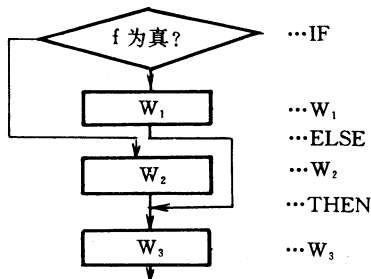
0 TEST

显示 This flag is false

1 TEST

显示 This flag is true

这个结构也可用框图说明



第⑧种结构也是分枝结构,是多路分枝结构,也叫 CASE 结构,与前两种相比,它不是用判别标志的方法,而是根据具体数据决定该做其中哪部分。

这个结构在 PC/FORTH 中和 D1.0 版本中有差别,这里先看看 PC/FORTH 版本中的 CASE 结构,在大多数版本中,都是这么用的。

```
(n) CASE
 n1 OF W1 ENDOF
 n2 OF W2 ENDOF
 n3 OF W3 ENDOF

 W0
ENDCASE Wn
```

这里 CASE 和 ENDCASE 类似于 IF 和 THEN,是这个结构的分界。整个结构要点是:

(1) 执行这个结构之前,先在栈中放一个被判断的数 n

(2) 当 n=n<sub>1</sub> 时,或者说 n 与 n<sub>1</sub> 匹配,执行 n<sub>1</sub> 后边 OF 和 ENDOF 中的 W<sub>1</sub>,然后退出这个结构,执行 W<sub>n</sub>。以此类推,n 等于 n<sub>2</sub>,执行 W<sub>2</sub>,然后退出,执行 W<sub>n</sub>。

(3) 如果 n 无法与 n<sub>1</sub>、n<sub>2</sub>、n<sub>3</sub>...相匹配,就执行 W<sub>0</sub>。但要注意,ENDCASE 会吃掉栈顶一个数,因此在执行 W<sub>0</sub> 后,应该在栈顶留下一个无用的数。

(4) CASE 与 ENDCASE 是必须出现的。OF 和 ENDOF 必须成对出现,但有多少对不限。

举一例,如果你正在学英语,对星期几在英语中如何表示还记不牢,就可以编一段小程序:

```
: OF_WEEK CR DUP . ." of week is
"
CASE
 1 OF ." Monday" ENDOF
 2 OF ." Tuesday" ENDOF
 3 OF ." Wednesday" ENDOF
 4 OF ." Thursday" ENDOF
 5 OF ." Friday" ENDOF
 6 OF ." Saturday" ENDOF
 7 OF ." Sunday" ENDOF
ENDCASE CR ;
```

这样定义之后,想知道星期三在英文中怎么说,就可以:

3 OF\_WEEK

显示 3 of week is Wednesday

对于 D1.0 版本,笔者也曾定义出象 PC/FORTH 中那样的结构,但想让这个结构适应范围更广些,于

是定义出第二种:

```
n CASE
 cd1 COND W1 CEND
 cd2 COND W2 CEND
 cd3 COND W3 CEND
 ELCASE W0
CASEND Wn
```

与前一种 CASE 结构的差别,主要是用 cd<sub>1</sub>,代替了 n<sub>1</sub>,cd<sub>2</sub> 代替了 n<sub>2</sub>...这里的 cd,笔者称之为条件,是诸如 5 > 0=这样的比较操作,以得出标志

当 n 符合 cd<sub>1</sub> 时,执行 W<sub>1</sub>

当 n 符合 cd<sub>2</sub> 时,执行 W<sub>2</sub>...

当 n 与所有条件都不相符时,就执行 ELCASE 之后的 W<sub>0</sub>。在执行 W<sub>0</sub> 时,栈中 n 已被清除,也不必在栈中留下一个数。

与 PC/FORTH 相似,CASE、ELCASE、CASEND 都是必需的。COND 和 CEND 必须成对出现,但多少对是不限的。

举一个例子,气象部门对小雨、大雨都是有其特定标准的,可以按 1 小时降雨量来衡量。

小雨 <=2.5mm

中雨 2.6~8.0mm

大雨 8.1~15.9mm

暴雨 >=16mm

于是就可以定义一个词,给出降水量,则输出雨量等级,这回使用汉语拼音。由于 D1.0 版本使用整数,这里要求输入数扩大 10 倍

```
: yu CASE
 25 <=COND ." xiao " CEND
 DUP 26 > SWAP 80 <= AND
 COND ." zhong " CEND
 DUP 81 >= SWAP 159 <=
 AND
 COND ." da "CEND
 160 >= COND ." bao " CEND
 ELCASE
CASEND ." yu " CR ;
```

定义之后,执行

23 yu↵

显示 xiao yu (小雨)

再输入

100 yu↵

显示 da yu (大雨)

象这个例子,查找一个数是否在某个范围,如果使用 PC/FORTH 中的 CASE 结构,就不太好办。可能使用 IF、ELSE、THEN 嵌套会更好些。

下一讲,将介绍另一类重要结构——循环。

## 初、中级程序员软件水平考试辅导

# C 语言试题与分析(续)

天津大学(300072)赵国瑞

例题 3[1992. 下午、六②]阅读下列程序说明和 C 程序,把适当的字句填入空格栏内。

[程序说明]本程序每次输入一个用户编码及其欠款金额后,就累计同一用户的欠款金额,并按用户编码由大到小的顺序排列已输入的所有欠款,最后输出排序后的全体用户的欠款总额清单。

[程序]

```
#include <stdio.h>
#define MN 500
long code[MN];
float money[MN];
main()
```

```
{int u,v,n;
 long t-code;
 float t-money;
 n=0;
 printf("Enter first code and money(-1 to quit)");
 scanf("%ld",&t-code);
 while(t-code>0)
 {scanf("%f",&t-money);
 u=0;
 while((1))
 u++;
 if((2))
```

```

 money[u] += t_money;
else
{for((3))
 {code[v]=code[v-1];
 money[v]=money[v-1];
 }
 code[u]=t_code;
 money[u]=t_money;
 (4);
}
printf("\n Enter next code and money (-1 to
quit)");
scanf("%ld",&t_code);
}
for(u=0;u<n;u++)
 printf("%5ld %f\n",code[u],money[u]);
}

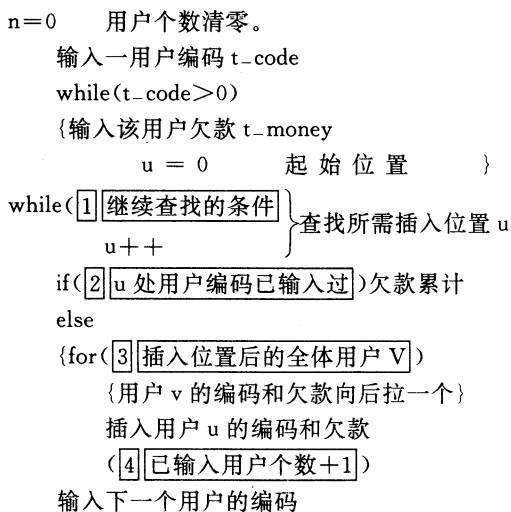
```

[分析与解答]这类试题中的程序说明部分是该题目的总纲。它包含着关于处理流程、处理算法以及题目难点和解答这些难点的提示性信息。所以首先必须认真阅读程序说明,从中提取解题所需要的关键信息。

从本题的程序说明可知,本程序涉及到两种算法,一是排序算法,二是求累加和算法。根据说明,因为每输入一个用户编码及其欠款后,就进行排序,判断所使用的是插入排序法。

紧接着,在阅读程序过程中,应该注意理解处理的大致过程。各语句组的功能。各变量的作用。在充分理解的基础上,集中思考各空格栏的位置及含义。按照它们的含义逐步完善所填内容。

对本题的处理大致过程,各语句组作用和各变量的作用的理解可用下图表示:



}  
打印结果

对于某些涉及 for 循环参数、赋值计算、输入/输出的空格栏,如本题的(3)和(4)可较快确定它们的内容分别为  $v=n$ ;  $v=u$ ;  $v--$ ;  $n++$ 。但是对于某些循环条件、选择条件的空格栏,如本题(1)和(2)在给出初步内容后,需仔细检查已列出条件是否完整。以空格(1)继续查找的条件为例,首先想到的条件是  $u$  处的编码应大于  $t\_code$ ,即  $code[u]>t\_code$ ,经过思考,查找过程应小于当前已输入的用户个数  $n$ ,故完整条件应写成  $code[u]>t\_code \ \&\& \ u<n$ 。

经过上述阅读程序说明,粗知所作处理、所用算法;阅读程序,理解处理过程、变量含义;填写空格栏处的内容并检查完善这三个步骤,可得到本题的最后答案为:

- (1)  $code[u]>t\_code \ \&\& \ u<n$
- (2)  $code[u]==t\_code \ \&\& \ u<n$
- (3)  $v=n;v>u;v--$
- (4)  $n++$

应当指出的是,这里所给的答案仅是满足题目要求的一种形式。此外还存在多种等价的答案形式,有时甚至是形式完全不同的答案。限于篇幅,本文未作这方面的分析。

例题 4(同例 3)

```

[程序]
#include <stdio.h>
#define MN 500
struct{
 long code;
 float money;
}debtor[MN], *u, *v;
main()
{int n;
 long t_code;
 float t_money;
 n=0;
 printf("Enter first code and money (-1 to quit)");
 scanf("%ld",&t_code);
 while(t_code>0)
 {scanf("%f",&t_money);
 u=&debtor[0];
 while((1))
 u++;
 if((2))
 u->money+=t_money;
 else

```



```

 {for((3))
 {v->code=(v-1)->code;
 v->money=(v-1)->money;
 }
 u->code=t_code;
 u->money=t_money;
 (4);u++
 }

printf("\n Enter next code and money (- 1 to
quit)");
scanf("%ld",&t_code);
}
for(u=&debtor[0];u<&debtor[n];u++)
 printf("%5ld %f\n",u->code,u->money);
}

```

相应的答案应为:

- (1) u->code>t\_code && u<&debtor[n]
- (2) u->code==t\_code && u<&debtor[n]
- (3) v=&debtar[n];v>u;v--
- (4) n++

本例中出现了结构及其指针。与例 3 相比较,由于结构指针的使用,把结构数组的下标隐藏起来了,因此,读起来不很方便,给解题增加了难度。但是只要多读多练,就会逐渐适应。为了对指针的应用增进理解,下面我们再看一个数组及其指针使用的例子。

例题 5 阅读下列程序说明和 C 程序,将适当的语句填入各空格栏内。

[程序说明]

已知某数列的前两项为 7 和 9,其后继项根据当前最后二项的乘积按下列规则产生:

- (1) 若乘积为 1 位数,则该数即为数列的后继项
- (2) 若乘积为 2 位数,则该数的十位数字和个位数字依次作为数列的两个后继项

本程序输出该数列的前 n 项以及它们的和。其中函数 sum(n,pa)返回数列的前 n 项之和,并将产生的前 n 项存于指针 a 所指向的数组中。

例如,若输入 n 的值为 12,则程序输出以下内容:

```

sum(12)=66
7 9 6 3 1 8 8 6 4 2 4 8

```

[程序]

```

#include <stdio.h>
#define MAXNUM 100
main()
{ int n, * p, * q,num[MAXNUM];
 do { printf ("Input N (2 < N < %d):",
 MAXNUM+1);

```

```

scanf("%d",&n);
}while((1));
printf ("\n sum (%d) = %d \n", n, sum (n,
num));
for(p = num,q = (2);p<q;p++)
 printf("%4d", * p);
printf("\n");
}

int sum(n,pa)
int n, * pa;
{ int total,count,temp;
 * pa=7;(3)=9;
 total=16;count=2;
 while((4)<n)
 { temp=(5);
 if(temp<10)
 {total+=temp;
 *(++pa)=temp;
 }
 else
 { *(++pa)=temp/10;
 total+= * pa;
 if(count<n)
 { count++;pa++;
 (6)=temp%10;
 total+= * pa;
 }
 }
 }
 }
 (7);
}

```

[分析与解答]对于结构成员的引用有三种方法:结构名,成员名,结构指针→成员名,( \* 结构指针). 成员名。最后这种方法已很少使用。对于数组成员的引用有两种方法:数组[下标表达式], \* 数组指针。应注意两者的差别。另外,本题中所涉及到的计算累加和与产生数列后继项两个算法都很简单,所以,只要熟悉 C 语言中数组指针的用法,就很容易得到如下答案:

- (1) n<3 || n>MAXNUM 或 n<3 || n>100
- (2) p+n
- (3) \* (++pa)
- (4) count++
- (5) \* (p-a) \* \* pa
- (6) \* pa
- (7) return(total)

# BJS-51 教学实验板的普及型新品 “DP-851 单片机普及板”技术讲座

北京市单片机应用技术协会秘书长 王玉民

编者 g 按:自本刊 92 年第 7 期举办 BJS-51 及 BJS-98 教学实验系统技术讲座以来,读者对这种公布技术细节的连载评价很好。为了满足众多“学装微电脑”爱好者的愿望,本刊将继续连载 DP-851 单片机普及板的介绍文章。该板已被中国计算机学会选作《单片机普及函授班》的教学实验设备,现介绍给读者,欢迎大家提出交流意见(信件请寄:北京市技术交流站单片机应用技术协会,邮编:100052)。

## 一、引言

DP-851 单片机普及板是一种为单片机普及、教学、开发服务的 51 系列单片机系统。北京市单片机应用技术协会为配合我国单片机技术教育的开展,集合了一批在单片机教学第一线长期从事教学的教师与长期从事单片机产品开发的工程技术人员,经过反复论证于九二年推出 BJS-51 及 BJS-98 教学实验系统(见 92 年 7~12 期及 93 年 1~4 期)。

随着我国计算机技术的迅速普及,涌现出很多学装微电脑的爱好者。为了使业余爱好者能够及早入门,并做到学用结合,解决实际问题,我们对 BJS-51 进行了优化设计。在保留 BJS-51 总体结构的基础上,使用新型薄膜键盘板和分离线路板设计技术,推出了物美价廉的“DP-851 单片机普及板”。现在该板已被中国计算机学会选作《单片机普及函授班》的教学实验设备,并已申请了专利(93301539.9)。

## 二、DP-851 单片机普及板的结构特点

- ① 体积小:主板 11cm×13cm+轻触键盘板 8cm×13cm(键盘覆盖在主板上),实验板 11cm×13cm(与主板软线连接),放入塑料包装盒中。
- ② 模块化:主板、键盘板、实验板可按需要灵活组合。如主板+键盘板(或主板+PC 机)=开发板;开发板+实验板=教学实验系统(配有教材)。
- ③ 可扩展:主板上 40 芯插头,可接 EPROM 固化板、四种 A/D 板、D/A 板、各种实验板等。
- ④ 实验多:实验板上有 IC 座、阻容件、发光、发声、驱动电路等分立元器件。实验时,使用“绕、褪导线工具(已申请专利)”把 0.25mm 的漆皮线绕接在与元器件连通的接线柱上,即可组成各种实验。该绕接方法比“面包板”插接可靠;又省却了用烙铁。

## 三、DP-851 单片机普及板的配置及功能

DP-851 单片机普及板是一种模块式结构的 51 系列单片机系统。它有三种配置方式以实现不同类型的需要。

### 1. 单版型自开发板配置

主板为一个带有自开发功能的单片机扩展系统。上有 16KB 的监控程序(键盘和 PC 机各占 8KB),具有置入、修改、调试、运行等多种开发功能。可用于对 51 系列单片机进行开发(需占用一部分用户资源)。

① 主板+薄膜键盘板,可构成单版型单片机自开发板。可以实现机器码的基本程序设计的编程及调试。

② 主板+PC 机,可构成 PC 联机型单片机自开发板。可以实现汇编语言的基本程序设计的编程及调试。

### 2. 基本配置(教学实验板)

基本配置是由开发板+实验板构成的教学实验系统。由于实验板上配置有进行实验用的接线柱(用绕、褪工具绕接)、LED 显示器、蜂鸣器、DIP 开关、发光二极管等,因此可以作 LED 显示、时钟、声音、交通灯、开关输入输出等常用教学实验。还可用主板上的六位 LED 显示器及键盘板上的 32 个键进行更为复杂的教学实验。该系统基本上能完成除 A/D、D/A 外的绝大部分教学实验。是一种大众型的教学实验设备。

### 3. 扩展配置(目标应用板)

开发板+A/D、D/A 等选件板可构成用户的各种目标系统。用户可通过 40 芯电缆连接各种应用板。目前已开发和正在开发的有 8 位 A/D、D/A 板,12 位 A/D 板、三位半及四位半 A/D 板,自升压 EPROM 固化板,语音实验板,电机实验板等选件板,用户可自行选择扩展。

以后各期陆续公布普及板的内部结构、原理图、监控程序框图、自升压 EPROM 固化板等技术细节。为便于学装微电脑的爱好者系统掌握单片机知识,下面把中国计算机学会与北京市单片机应用技术协会联合举办的“单片机普及函授班”招生简章公布如下,请参考。

## 招 生 简 章

为在我国青少年和科技人员中普及计算机硬件、软件知识,推动单片机的普及应用,计算机学会普及工作委员会在举办多期“MP-1型学习机”函授班的基础上,准备将“学装微电脑”的普及活动档次升级,让学习者使用单片机解决实际问题。为此,中国计算机学会与北京市单片机应用技术协会决定联合举办“单片机普及函授班”。每年举办两期(春季、秋季),并根据条件与可能,委托地方学会开办函授点。举办函授班的宗旨是,扩大青少年和科技人员的知识面,增强他们的动手能力,以适应社会对新技术、新知识的需求。并计划于九四年底举办“学装微电脑”的全国比赛(进行现场和成果比赛),为将我国的计算机普及活动推向新高度作出不懈的努力。

### 一、招生对象

具有中等以上文化程度的微电脑爱好者、中学师生、各行业的技术人员、校内外科技辅导员、对单片机应用开发有需要的或有兴趣的业余爱好者等。

### 二、教学计划与学习要求

每年举办两期函授班(春季和秋季)。每期函授班共四个月,分为四个教学段进行学习。

向每位学员提供学习所需的全部材料,并配备一套“DP-851单片机普及板”作为教学实验设备(含主板、键盘板、实验板、电缆、工具、电源、盒、说明书)。

要求学员按各阶段的教学安排寄交作业及实验报告,并提出疑难问题。由指导教师批改作业,解答问题,寄给学员。学习结束后举行开卷考试,按作业和考试成绩发给“全国单片机普及技术函授班结业证书”。

### 三、报名和开学日期

春季班报名从1月1日起到3月15日止,4月1日前寄出教材及实验设备。

4月15日开课,8月15日结业。

秋季班报名从5月1日起到7月15日止,8月1日前寄出教材及实验设备。

8月15日开课,12月15日结业。

### 四、教材及学费

全部学费398元,包括教材费及实验设备费在内。教材包括《单片机普及函授班教材》上、下册,由清华大学吴文虎教授和北京广播电视大学李广弟副教授执笔(共二十多万字)。

实验设备使用北京市单片机应用技术协会研制的“DP-851单片机普及板”,提供《DP-851单片机普及板使用说明书》一册。学习结束后,该设备将成为您进行单片机开发应用的得力工具。

### 五、报名办法

报名者可写信到中国计算机学会办公室,索取“学员登记表”。学员填好后连同学费一并交齐。函授班将发出“录取通知书”及学习用全套教材、教学大纲和教学实验设备。

地址:北京2704号信箱,邮码:100080,联系人:宁伟成,

注:第一期函授班从即日起开始报名,1993年8月15日开课。

## SCB系列单片单板机介绍

SCB系列单片单板机是武汉尚吉电子研究所推出的一种以单片机作CPU的单板机,它的最大特点是集教学、开发、应用于一体。其中,SCB-1型已销售近两年,取得了广泛良好的社会效益,众多的用户已用它开发成许多产品。现在,SCB-2型机又隆重推出,其各项性能均有新的突破。分别介绍如下:

SCB-1型以8031为CPU,系统时钟最高可达12MHz,扩有8K EPROM、带掉电保护的16K RAM、6位LED显示器、5×5键盘及“看门狗”电路,还扩有8

路8位A/D,1路8位D/A,并配有一个标准RS232口,及由8255扩充的3个8位并行或位控I/O口,还可直接选配8K或16K EEPROM片子。除以上硬件配置外,SCB-1还有逾4K的监控程序,它包括单步执行、现场保护与恢复、键扫描、显示、存储器读写、存储器数据移动、串行通信接收与发送、A/D转换、打印机驱动及检空、比较、定时时钟等众多子程序。

SCB-1的监控程序可以使用户通过键盘操作进行搬家、检查、改写内外RAM并可连续或单步执行程序



序,直接与 PC 进行通信,固化 EPROM 等操作,除监控软件外,还配一个在 PC 机上运行的软盘文件,它有正反汇编、正反通信、编辑及监控程序清单。为更好地推广应用这一产品,本所在武汉大学出版社以大中专教材形式出版并发行了《单片机原理及制作》一书,该书对 SCB-1 的全部软硬件作了详尽介绍,并配有实验操作及应用实例等。全书约 40 万字,定价 8.5 元(含邮资)。

新推出的 SCB-2 型则是在 SCB-1 型上的再次提高,SCB-2 型单片单板机集 51 及 98 系列于一机,是教学领域之佼佼者,开发及应用的最佳选择硬件上,它配有 8031 和 8098 两个 CPU,除保留 SCB-1 的全部硬件外,新改 8 位 LED 显示,双复位键,键盘及显示改为 8279 驱动,有薄膜面板按键的接口,显示及键盘部分更为小巧。ROM 改为 16K,整机由单一+5V 电源供电,配有掉电保护电路及可充电电池,可断电保持数据达 3 个月以上,扩展引出线除保留同 SCB-1 兼容外,新增 60 个引出线,将 8031 及 8098 的总线及所有口全部引出,供用户选择。本所另备有与 SCB 系列单片单板机配套的仿真板,利用本仿真板可实现 SCB-2 型单片单板机的一次性零地在仿真功能。加上单板

单位:武汉市尚吉电子研究所 地址:武汉武昌珞瑜路 37-3 号  
电话:716138 716692 电报挂号:4561  
邮编:430070 联系人:单部  
深圳办事处地址:深圳市深南中路 75 号科技商场 47 号柜  
邮编:518031 联系人:罗胜辉 李秋华

机本身具有的单步、断点、修改及其通信、固化等功能,构成一套完整的开发系统。

SCB-2 型的监控程序设计更为完善,它有 8031 及 8098 两套监控,监控程序除保留原 SCB-1 型的全部功能外又新增了 INS(插入)及 DEL(删除)等功能,并新增了 CRT 监控功能,即通过 PC 机(或 286、386)即可实现对 SCB-2 型上的程序进行调试,搬家、检查、修改内外 RAM、单步连续运行程序,显示执行结果等,现改现调,使用非常方便。

随 SCB-2 型配有软盘 2 张,分别用于 51 及 98 系统。

本所诚为推广单片机技术的应用,毫无保留地将 SCB-2 型的软硬件资料全部公开给用户使用。并配有详细的使用说明,随《SCB-2 单片单板机详解》一书一并详尽介绍。

销售服务方面本所本着社会效益及经济效益并重之原则,SCB-1 型定价为 495 元/台(整体式)535 元/台(分体式),SCB-2 型零售价为 795 元/台。本所备有详尽功能介绍及销售服务说明,函索即寄,欢迎与本所在全国的经销代理或本所联系。

## 从三种通用的八位单片机硬、软件 比较来看 Z8 系列单片机的优越性(续)

Zilog 中国区高级工程师 董伯明

9. 串行通信接口 Z8611 有一个可编程的串行通信接口,该芯片内含一个 UART,完成全双工的异步串行接收/发送工作,由 0 号计数器/定时器控制传输速率,最大可适 93500b/s。当一个汇编字符被送到接收器的缓冲区时将产生一个中断,发送字符产生一个分离的中断,接收寄存器是双缓冲,附有硬件的奇偶校验和检测。

Z8051 是利用一个并行端口来完成串行 I/O 功能,由 1 号计数器/定时器完成速率控制,最大传输率可达 187500b/s,当完成一个发送/接收产生一个中断,接收寄存器亦是双缓冲。

MC6801 内含一个全双工异步串行通信接口。位传输率控制是由一个速率寄存器和 MCU 的时钟或外部时钟组成,最大传输速率是 62500b/s。发送/接收寄

存器都是双缓冲的,同 8051 一样完成一个发送/接收产生一个中断,无硬件的奇偶校验与检测,对数据出错和溢出运行可以自动检测。

对 8051 和 MC6801 完成一个发送/接收后才产生一个中断。但 Z8611 分别产生一个中断,这能增强利用串行通信的能力,因为在发送/接收时常要用到独立的服务子程序。8051 的传输速率高于 MC6801,Z8611,然 Z8611 有一个硬件的奇偶校验,通过二条简单的指令就能完成数据出错和溢出运行的检测。

### 二、指令系统:

Z8611 的指令系统是为 MCU 的特殊应用场合而设计的,故而无论在软件设计及最后程序运行的优越性,在后面的三种 MCU 的软件运行比较中可明显地

体现出来。

1. 寻址方式 Z8611 和 8051 都有 6 种寻址方式: 寄存器、间接寻址、变址、直接、相对与立即。MC6801 只有五种: 累加器、变址、直接、相对与立即。从表一可看出, MC6801 的寻址方式受到很多限制; 8051 虽然具有 Z8611 的同样的六种寻址方式, 然而亦受到很多限制, 而 Z8611 的每条指令具有寻址方式的很多优越性主要在于它的每个寄存器都可当作累加器使用这个特殊的结构、特殊的优越性。举个简单的常例: 如清除、求补、移位, 四位交换等常要用到累加器的指令在 8051 中只能在累加器中完成。而在 Z8611 之中, 只要用寄存器寻址或间接寄存器寻址就可以在任何一个寄存器中完成。

2. 变址寻址 三种芯片寻址方式区别最大的是变址寻址。Z8611 可用任一条寄存器作变址; 而 8051 只能用累加器作变址, 而且要加上数据指针和程序计数器; MC6861 有一个 16 位的变址寄存器, 存放在指令第二字节的地址被附加到变址寄存器的低八位, 而进位位被加到高八位, 这样才完成了变址。MC6801 只有一个 16 位的变址寄存器和从指令第二字节取来的 8 位立即数。8051 只有一个累加器可作变址用, 限制了数据流的传送。

3. 短寻址方式和长寻址方式 短寻址能节省内存和提高运行速度。三种芯片都有短寻址方式, Z8611 在外接存储器和内部寄存器中均可采用短寻址方式。8051 只能在最低 32 个寄存器中使用。

Z8611 有二种寄存器的寻址方式, 整个八位作为地址, 或用寄存器指针形成四位地址。对 Z8611 内部的 256 个寄存器可分成 16 个组, 每组 16 个寄存器, 寄存器指针 RP 完成指出 16 个组中的一组, 而每一组中的 16 个寄存器地址, 只需 4 位就能完成寻址。在外存的寻址, 原来需 16 位, 现在用 12 位就能取代了; 用一个端口的 8 位地址/数据复合总线 and 另一个端口的 4 位组成了一个 12 位的寻址, 而余下的另一端口四位可以作 I/O 口, 此特性可允许最大寻址到 10K。

8051 只能在最低 32 个寄存器采用这种方式。它将 32 个寄存器组成 4 组, 通过 PSW 中的 2 位来选中某一组, 而采用指令中的 3 位来寻到每一组的每一个寄存器。

MC6801 采用一种增强型的寻址方式来寻外部存储器, 采用一个特殊的非多重扩展模式, 外部存储器的 256 字节可以存取, 它用一个八位端口作地址, 另一个八位端口作数据。

4. 堆栈 Z8611 和 MC6801 可采用外存作堆栈, 此时有一个 16 位的堆栈指针 SP。采用内部寄存器作堆栈时, 只需一个 8 位的堆栈指针。而 8051 只能用内存作堆栈, 有一个 8 位堆栈指针。总之, Z8611 由于寄存器阵列的灵活便利, 寻址方式的特点, 尤其变址的能力和短寻址方式, 存储器空间的利用率和指令执行时间的节省使其性能高于 8051 和 MC6801。

### 三、开发支持

现从芯片, 软件和仿真等方面来讨论三个品种的 MCU 的开发能力。

1. 芯片 Z8 单片机系列有多种芯片来支持产品的开发。Z8611 是一个带有 4K 掩膜 ROM 的单片 MCU。对其开发可有三个芯片来支持, Z8612 是 64 脚的不带 ROM 的开发芯片; Z8613 是背驮式的可开发 Z8611 的另一种芯片, 它可插上 4K EPROM(2732) 或 2K EPROM(2716)。第三种芯片 Z8681 是 40 脚的, 其特性和 Z8611 完全一样不带 ROM。

Intel 提供的 8051 有不带 ROM 的 8031 与内带 4K EPROM 的 8751。Motorola 提供的 MC68701 有 2K 的 EPROM; MC6801 有 2K ROM; MC6803 无内部 ROM; MC6803NR 内部不带 ROM 和 RAM。

2. 软件 由于 MC6801 与 6800 兼容, 故不需汇编。Z8611 和 8051 均需汇编。

3. 如今三种芯片均有开发、仿真系统了, 而 Z8611 的支撑芯片更方便其开发, 仿真。

(待续)

## 与 CEC—I 微机接口的 EPROM 仿真卡

宁夏教育学院教务处(750002) 王正英

在单片、单板机的应用中总是要使用开发系统。从开发系统的工作过程来看, 主要是通过 CPU 在线仿真来完成开发任务。CPU 在线仿真时通过运行暂存在开发系统中的目标程序, 对目标机的各项功能进行检验, 对目标程序进行纠错、调试, 最后将调试好的目标程序固化在 EPROM 中, 插在目标机程序存储器的插座

上, 开发工作宣告结束。由此看来开发工作的重点之一是在固化目标程序之前, 使其在目标机上运行, 以检验目标程序是否能够达到预期的目的。如果直接将目标程序写入 RAM 中, 再通过仿真插头将 RAM 接到目标机程序存储器的 EPROM 插座上, 这样目标机就可以运行未固化的目标程序了, 这就是所谓的 EPROM

—EPROM 仿真。本文介绍的 EPROM 仿真卡就是根据这一思路设计的,可以仿真 EPROM2716~27256,具体的电路如图 1 所示。如果该卡配合使用相应的汇编,及汇编工具软件,实际应用时将会更加方便。

硬件结构:图 1 电路主要是一个 8255 芯片和一个 65256RAM (32K×8 位)芯片。8255 的 A 口与 65256 的 8 根数据线相连,用来传送数据。B 口用来产生 65256 的低 8 位地址信号。C 口的 C<sub>0</sub>~C<sub>6</sub> 用来产生 65256 的高 7 位地址信号,C<sub>7</sub> 通过一个非门产生  $\overline{RD}$  和  $\overline{WR}$  信号。65256 的  $\overline{CE}$  端通过 KA 接地,这样用一片 8255 就能完成对 65256 的读写,图 1 电路采用了备用电源。关机后备用电源供电,所以 RAM 中的数据不会丢失,此时将仿真卡从微机上拔下可进行脱机仿真。8255 与 CEC-I 微机接口时占用四个地址,A 口的地址为 \$C090(49296),B 口的地址为 \$C091(49297),C 口的地址为 \$C092(49298),控制寄存器的地址为 \$C093(49299),向 65256 写入数据时,应写入控制寄存器的内容(控制字)为 \$80(128);从 65256 中读出数据时,应写入控制寄存器的内容(控制字)为 \$90(144)。8255 芯片可方便地与多种微机接口,只要编制相应的软件,该卡可以在多种微机上使用。

软件:为了达到界面好,速度快的目的,控制该卡工作的软件由两部分构成,一部分是用 BASIC 语言编写的引导程序(见程序 1),用来置初值,完成 EPROM

#### 程序 1

```

10 D$=CHR$(4);B$(4)="2 "
20 IF PEEK(6)=5 THEN 40
30 PRINT D$"BLOAD F8255,A$300"
40 POKE 6,5
50 POKE 49299,128;W=0;Q=0
60 HOME;PRINT "1---2716";PRINT
70 PRINT "2---2732";PRINT
80 PRINT "3---2764";PRINT
90 PRINT "4---27128";PRINT
100 PRINT "5---27256";PRINT
110 PRINT "6---END";PRINT
120 INPUT "1,2,3,4,5,6->?";I
130 IF I>6 THEN 50
140 ON I GOTO 150,160,170,180,190,270
150 S=120;D=2716;B$(1)="OF
 ";B$(2)="OF ";B$(3)="OF ";GOTO 200
160 S=112;D=2732;B$(1)="ON
 ";B$(2)="OF ";B$(3)="OF ";GOTO 200
170 S=96;D=2764;B$(1)="ON
 ";B$(2)="OF ";B$(3)="OF ";GOTO 200
180 S=64;D=27128;B$(1)="ON
 ";B$(2)="ON ";B$(3)="OF ";GOTO 200
190 S=0;D=27256;B$(1)="ON
 ";B$(2)="ON ";B$(3)="ON ";GOTO 200
200 POKE 239,255;POKE 238,127;POKE 235,S+W;
 POKE 236,Q
210 HOME;VTAB 10;HTAB 15;PRINT "KA->1";GET

```

#### A\$

```

220 CALL 768
230 HOME;VTAB 10;HTAB 15;PRINT D;PRINT
240 HTAB 10;PRINT"K1 "; "K2 "; "K3 "; "KA ";PRINT
250 HTAB 10;PRINT B$(1);B$(2);B$(3);B$(4)
260 GET A$;GOTO 50
270 HOME;VTAB 10;HTAB 15;PRINT "[END]";END

```

型号的选择、操作提示、调用主程序等任务,另一部分是用汇编语言编写的主程序(见程序 2),主要完成将目标程序写入 65256RAM 的任务。为了使该卡能适应 2716~27256 EPROM 仿真,在内存安排上将 CEC-I 微机中 \$1000~\$9000 共 32KB 的空间让给目标程序占用,由于引导程序不长,主程序短小(只有 65 个字节),且从 \$300 开始存放,所以程序运行时不会引起混乱。65256 的地址信号 B<sub>0</sub>~B<sub>7</sub>,C<sub>0</sub>~C<sub>6</sub> 由低位向高位产生,首地址在程序 1 中固定为 \$0000(见程序 1 的 50 句),在使用中若想变动,可将 50 句改为 INPUT 语句,在 \$EB(235)中存放地址的高位,\$EC(236)中存放地址的低位。

#### 使用方法:

1. 首先将程序 1 送入计算机,然后将程序 2 以 \$300 为首地址键入,并以 F8255 为名存盘。

#### 程序 2

```

0300- A2 00 LDX # $00
0302- A5 EB LDA $EB
0304- 8D 92 C0 STA $C092
0307- 85 ED STA $ED
0309- A5 EC LDA $EC
030B- 8D 91 C0 STA $C091
030E- BD 00 10 LDA $1000,X
0311- 8D 90 C0 STA $C090
0314- A5 ED LDA $ED
0316- 09 80 ORA # $80
0318- 8D 92 C0 STA $C092
031B- A5 EF LDA $EF
031D- C5 EC CMP $EC
031F- D0 11 BNE $0332
0321- A5 EE LDA $EE
0323- C5 EB CMP $EB
0325- D0 0B BNE $0332
0327- A9 00 LDA # $00
0329- 8D 0F 03 STA $030F
032C- A9 10 LDA # $10
032E- 8D 10 03 STA $0310
0331- 60 RTS
0332- E6 EC INC $EC
0334- D0 02 BNE $0338
0336- E6 EB INC $EB
0338- E8 INX
0339- D0 07 BNE $0302
033B- EE 10 03 INC $0310
033E- 4C 02 03 JMP $0302

```

2. 在关机的情况下将仿真卡与 CEC-I 微机的外设槽口(1号) 接好。

3. 开机后先将目标程序以 \$ 1000 为首地址调入内存。

4. 运行程序 1, 根据屏幕提示选择待仿真的 EPROM 的型号, 将  $K_A$  拨向“1”, 计算机会自动把目标程序装入 65256RAM。

5. 根据屏幕提示将  $K_A$  拨向“2”, 并预置好  $K_1 \sim K_3$  的状态,  $K_1 \sim K_3$  的状态与不同型号的 EPROM 的对应关系如表所示。

| 型号    | $K_1$ | $K_2$ | $K_3$ | $K_4$ |
|-------|-------|-------|-------|-------|
| 2716  | OFF   | OFF   | OFF   | 2     |
| 2732  | ON    | OFF   | OFF   | 2     |
| 2764  | ON    | OFF   | OFF   | 2     |
| 27128 | ON    | ON    | OFF   | 2     |
| 27256 | ON    | ON    | ON    | 2     |

6. 将仿真插头插在目标机的 EPROM 插座上即可进行 EPROM 仿真。

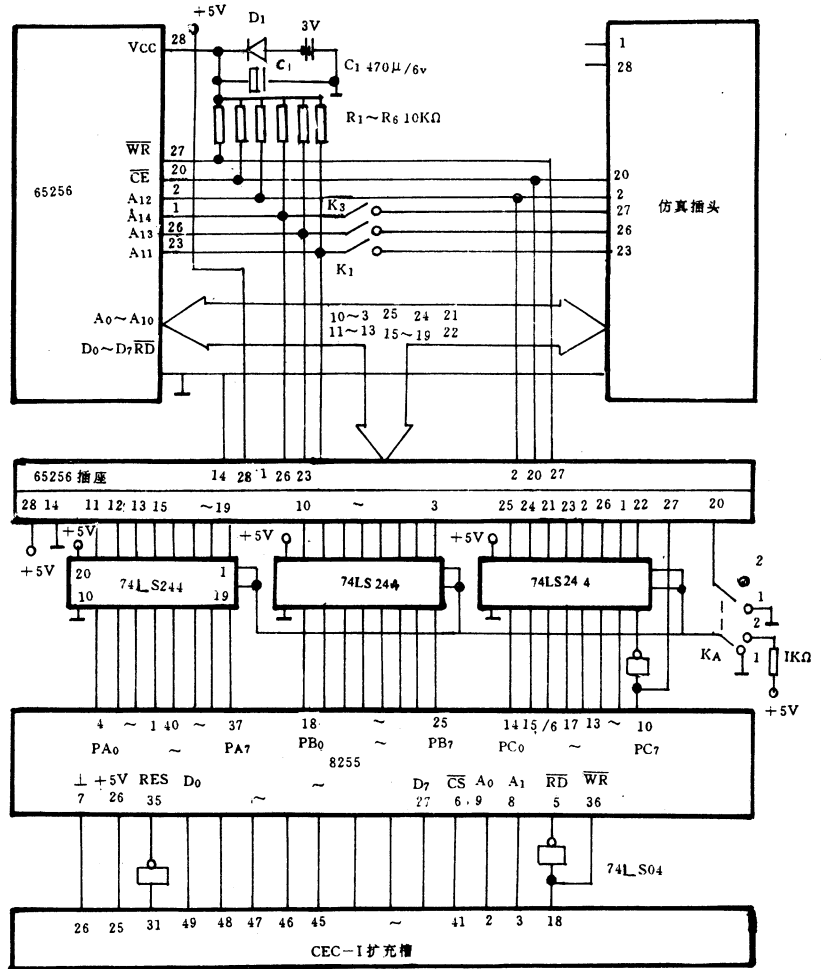


图 1

(接 40 页)

本文在简述彩显工作原理的基础上, 介绍了工作实践中遇到的非一般性常见故障的分析诊断和处理过程。这种方法也是常规方法, 即从输出模块一级级往回查, 直到找出故障。当熟知了彩显的工作原理且积累了一定的经验之后, 就不一定要采取上述常规方法了。如“场扫描电路出现的故障”一章中的例 2, 若抓住场偏转线圈上的锯齿波电流随时间推移频率变高, 幅度下

降这一主要特征, 就可以省去中间测试环节, 直接测 HA11423③脚的波形, 这样故障点一下就找到了, 省了许多时间, 给用户提供了方便。本文中有好多例子均可以用非常规方法诊断分析。但作者为了让读者尽量多的了解不同类型和牌号的彩显的工作过程(尽管大同小异), 用常规的方法、较长篇幅给予了解释。愿本文对读者有所帮助。



# PC 机并行口的扩展

武汉市海军工程学院(430033) 孙胜春 邢昌凤 肖英

本文将向广大 PC 机用户介绍一种既经济又实用的并行口扩展板电路。该电路经我们长达两年多的使用,工作性能可靠,操作使用方便,适用于 PC/XT、PC/AT 以及与之相兼容的其它机型。

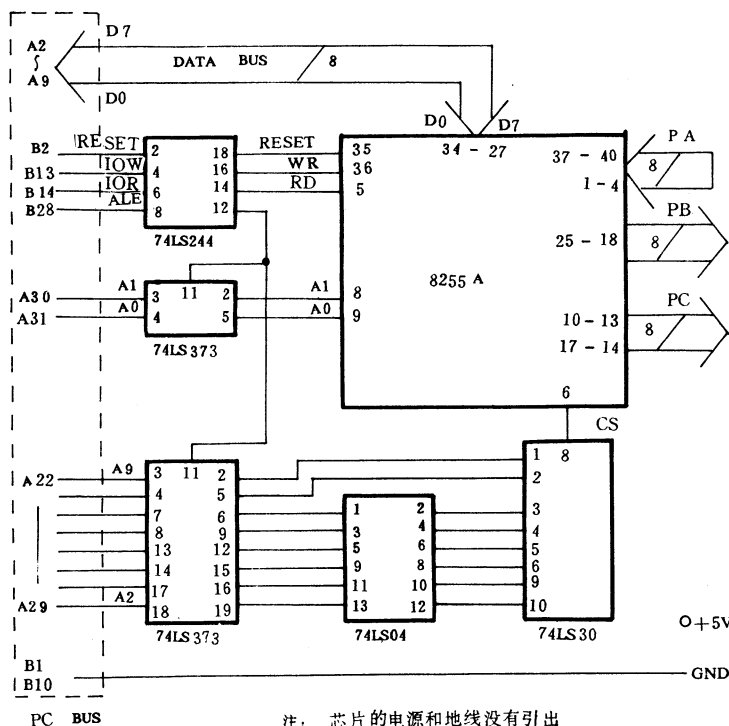
目前国内 PC 机用户较多,且大部分用户已用于实时采集处理、生产过程控制、用户系统管理等方面。

我们在实际应用中发现 PC 机只配有一个并行口,且已接成打印机使用状态,若要做其它 I/O 使用,需对其改造才成,对用户造成较大困难。现向用户介绍一种线路简单,易于实现的 PC 机并行口扩展电路,做为 I/O 扩展使用。可插入机内扩展槽内,永久使用。

## 一、原理线路实现

本文介绍的并行口扩展线路是基于 I/O 扩展基础上完成其功能实现,它只使用了 PC 机标准的 62 脚总线,故制作的插件板不仅适用于 PC/XT 总线要求;同时也适用于 PC/AT 总线。具体原理线路如下:

## 二、电路使用及说明



电路中的 244 芯片是一个起缓冲/驱动作用,从计算机总线上引出的控制信号由它控制电路的读/写和复位。两片 373 芯片主要用于在电路 I/O 操作期间将总线引出的 A<sub>0</sub>-A<sub>9</sub> 地址线进行锁存和整形,以保证 I/O 操作协调可靠。其中 373 的 11 脚是由总线 ALE “允许锁存信号”进行控制,其 1 脚接地。电路中的 04 和 30 芯片主要起口地址译码作用。本电路使用的 I/O 空间地址是机内保留的 300H-31FH 中的四个地址。具体分配如表:

| 口 名 称 | 口 地 址 |
|-------|-------|
| PA 口  | 300H  |
| PB 口  | 301H  |
| PC 口  | 302H  |
| 控制字口  | 303H  |

另外,电路中芯片不用的输入脚均接地。+5 电源是外加的;也可与 PC 总线上的 B<sub>5</sub> 和 B<sub>29</sub> 脚相连。但使用便携机的用户最好采用外加形式,以减轻主机电源之负载。电路中的地线必须与主机相连。用户在制作印刷线路板时,每块芯片上的电源与地线之间需加 0.1μF 的独石电容一只进行去耦,以保证电路的可靠工作。

## 三、结束语

本电路是我们结合课题需要研制的。由于 8255 并行口控制操作方便,组合使用灵活;所以在实际应用中控制多达五种类型十五个通道的输入/输出。其初始化程序只要一条即可完成对 8255 三个口任务的分配。本例中令 PA 口为输入;PB 口和 PC 口为输出方式,其控制字命令为:

OUTPUT ( 303H ),  
90H

# 彩色显示器工作原理简述及故障诊断(下)

中国人民大学信息中心(100872) 胡野红

## 四、场扫描电路出现的故障

场扫描电路的作用就是给场偏转线圈提供线性良好的锯齿波电流使之产生均匀变化的垂直磁力。当显像管阴极发射的电子束经过偏转点时,在此力的作用下,电子束产生从上到下的扫描。场扫描电路和行扫描电路相比,虽然功率小,频率低,但毕竟是功耗电路,所以故障常有发生。当场扫描电路出现故障时,其典型故障现象为水平一条亮线。

例 1:故障机型号:Compaq 牌 EGA 彩显

故障现象:水平一条绿亮线

分析诊断:从故障现象可以看出,场扫描电路因某种原因没工作,顺着场扫描电路的输出端场偏转线圈查,发现场扫描电路使用一块专用集成电路 TDA1170,如图 1 所示。测场偏转线圈  $V_{DY}$  的电阻值和 C216 的容量,未发现明显的损坏。上电测得  $V_{DY}$  端电压为零伏, TDA1170 ④端(场输出端)为零伏。由此可以看出 TDA1170 没工作。出现这种情况无非是 TDA1170 损坏或电源有问题。TDA1170 损坏的概率极小所以先不怀疑它。上电测得集成电路电源端②脚为零伏,说明 TDA1170 因无工作电压源而没工作,继续查找发现电源端限流电阻 R220(8.3 $\Omega$ /1W)上端有电压,而下端没有。再细看,R220 有明显糊状,拆下该电阻一量,果然开路,换上同等指标的电阻上电试验,屏幕扫描光栅展开。但屏幕上仍有绿色回扫线。此现象说明还有故障存在。

屏幕上有绿色回扫线且连机屏幕上不出字符,表明视放级或消隐电路有问题。因本故障是绿色回扫线,所以先检测尾板上 G 极电压(视放级输出),测量结果是 G 极电压仅为 3 伏,大大低于正常值 96 伏。测视放管 Q505(2SC1515)的各极电压,发现集电极无电压,顺着 Q505、Q511 往回测,如图 2 所示,测得 R526 上端有电压、下端无电压,断电拔下尾板,焊下 R526、果然 R526 开路,换上相同参数的电阻上电联机,显示器工作正常。

从以上过程可以看出,由于给场扫描集成电路供电的限流电阻 R220 开路, TDA1170 得不到工作电压而无输出,场偏转线圈中无电流通过,不能产生场扫描所需的垂直磁力,所以尽管行扫描电路能正常工作,但因无垂直洛伦兹力,使垂直扫描不能拉开,屏幕上只能是水平一条亮线。排除了场扫描集成电路 TDA1170 供电端的限流电阻 R220 开路的故障以后,扫描光栅虽然拉开,但屏幕上仍有绿色回扫线出现,根据彩显的

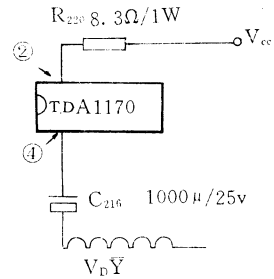


图 1

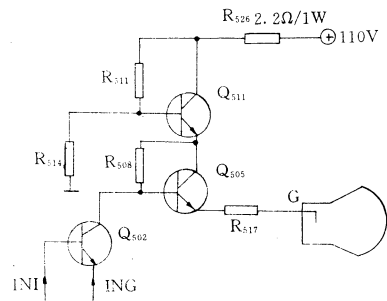


图 2

工作原理,查视放级的输出端,即显像管的阴极 G,由于 G 极电压大大小于正常值,故往前查视放管 Q505,据 Q505 集电极无电压这一现象,查得限流电阻 R526 开路。由于视放管得不到工作电压,显像管阴极 G 就不能发射电子束。所以,在扫描逆程,视放管该截止而不截止,又不能发射电子束,所以屏幕上出现绿色回扫线。

例 2:故障机型号:Texas 牌 CGA 彩显

故障现象:开机后屏幕上有光栅,但垂直方向上下各有 1/5 黑屏,扫描光栅呈宽银幕状。

分析诊断:从故障现象看,本机场扫描电路虽能工作但不正常。针对该机场扫描电路常出故障的特点分场输出级和场振荡级两部分讨论。

场输出级由 Q410、Q402、Q421、C402、C406、

C407、C419 等组成。如图 3 所示。这也是场输出级电路极易出问题的元器件。从场电路虽能工作但不正常看出,可能是输出功率不足或频率不对造成“宽银幕”现象。不排除上述元件性能变差的可能性。

用替换法对本机上述易变质的电容进行代换,故障现象依旧。再焊下 Q401、Q402、Q421 测试,也没发现异常现象,场输出级故障的可能性先排除。

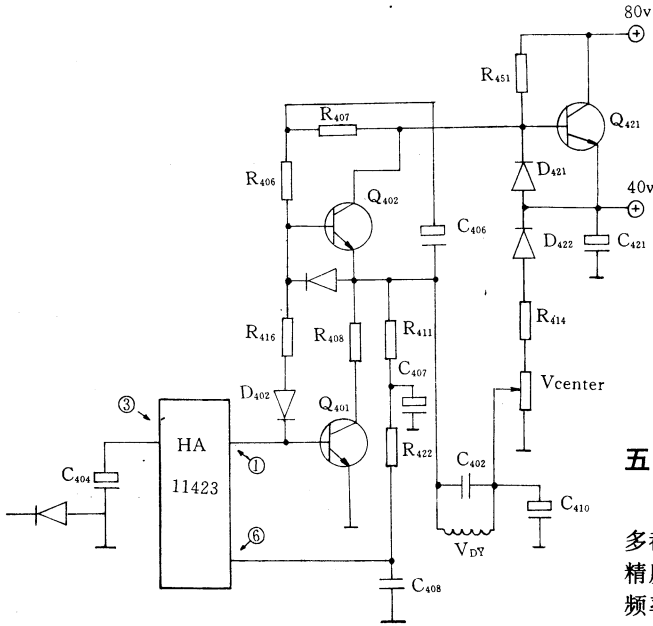


图 3

用示波器看偏转线圈上的波形,发现锯齿波的幅度随时间推移由 10 伏逐渐减小为 3 伏,周期也随之变小。由于场输出级不存在电容漏电、三极管性能变差等问题,所以故障肯定在场振荡级。

本机行、场电路共用一块集成电路 HA11423,在场振荡输出端①脚用示波器看,幅度和周期如前所述。可以肯定故障出在 HA11423 本身及其周围电路,用示波器测 HA11423③脚(场振荡)发现该点波形和场偏转线圈上的变化相同。由此更证明了前述判断的正确性。即故障在场振荡级。本级的核心 HA11423 损坏的可能性极小。先假设排除其损坏的可能性。既然周期有变化,就查和频率有关的钽电容 C404(25v/1μF)。焊下测量,容量明显变小,换上相同指标的钽电容上电试机,宽银幕现象消失。显示器工作正常。

正常工作时场偏转线圈和 HA11423③脚的波形如图 4、图 5 所示。

以上分析可以看出,由于场振荡电容 C404 性能变差,容量随温度增高而减小,致使振荡频率变高,幅度下降,产生“宽银幕”现象。

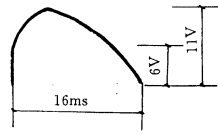


图 4



图 5

## 五、电源电路出现的故障

为了得到高质量的直流电压,显示器电源大多都采用开关电路。该电路的特点是功耗小,输出电压精度高,具有完善的可靠的保护线路。但因其工作在高频、高电压大电流的环境中,若机房环境不好,如过于潮湿,温度偏高,灰尘过大都容易造成显示器电源损坏,所以显示器电源故障在整个显示器故障中仅低于行扫描电路。

例 1:故障机型:IBM 牌 CGA 彩显

故障现象:指示灯不亮,无扫描光栅

分析诊断:电源指示灯不亮有两种情况,可能是负载有问题引起电源执行保护,也可能是电源本身损坏。打开机器断开负载,(220V/60W)用灯泡作假负载,上电后无电压输出(灯泡不亮),以上试验说明电源本身有故障。若有两台型号相同的机器,可以用替换法区分是负载故障引起电源不工作还是电源本身有问题

先不上电测保险管 F 和整流桥 D<sub>1</sub>~D<sub>4</sub>、开关管 Q801(如图 6 所示)的在线电阻,发现整流桥、Q801 明显损坏,换上相同技术的指标的整流桥和开关管后上电一试,机器工作正常。

根据损坏的元件分析其损坏原因,发现原整流桥耐压仅为 250 伏,损坏现象为其中的一桥臂短路。当整流桥一臂短路后,其输出电压就要大于 310 伏,此电压足可以击穿 Q801 使电源不能正常工作。由于电网电压波动大,整流桥耐压显得太低,所以换上耐压 300 伏的整流桥后,该电源的薄弱环节得到改进,在这种牌号的显示器上常发生的整流桥、开关管损坏现象再也没发生过。

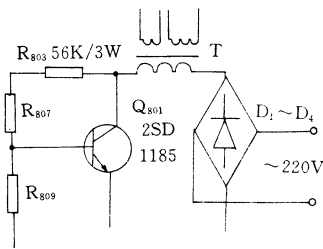


图 6

**例 2:故障机型:Solar 牌 CGA 彩显**

故障现象:电源指示灯不亮

分析诊断:从图 7 可以看出,本机是有两路输出的它激式开关电源,它激励信号从 TDA4601②脚输出。断电测开关管 2SC1942 和整流桥等易损件的在线电阻,未发现明显的开路短路现象。

排除负载发生故障致使电源不能正常工作的可能性之后,上电测得整流输出端有 310 伏直流,开关管 2SC1942 集电极有 310V 电压,而基极无驱动脉冲。怀疑 TDA4601 或过流过压保护线路有问题,换上 TDA4601 后故障依旧,其保护电路也未发现异常现象。上电测该集成电路各脚电压,发现电压几乎均为零。所以怀疑供电线路有故障。

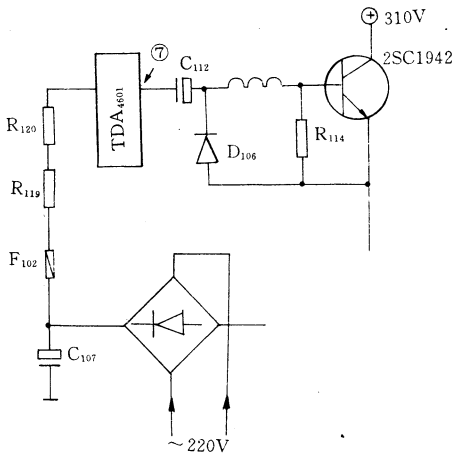


图 7

从整流桥 310 伏端查起,经过直流保险 F102、R119、R120 发现 R119 上端有电压,而 R120 上端即 TDA4601④脚无电压,焊下 R120(100k 1/2W)测试,果然开路,换上相同规格电阻后加电试机,显示器工作正常。

本例中由于它激脉冲形成集成电路 TDA4601 的

供电限流电阻 R120 开路,集成电路不工作,2CS1942 得不到它激脉冲、造成机内开关电源无输出故障。

**例 3:故障机型:AST 牌 VGA 彩显**

故障现象:电源指示灯不亮

分析诊断:本电源有 3 档输出,分别为 150V(白)63V(红)、15V(黄)。在排除负载损坏引起电源不工作的可能之后,由图 8 可以看出,开关管 Q901 (2SC3153) 的基极脉冲来自 WPC1394、而 Q904 (TIP50) 则组成保护执行机构。一旦发生过流过压, Q905 则导通, Q901 基极它激脉冲被短路,电源就停止工作。

断电测主要元件的在线电阻,发现 Q901 正常, Q905 有问题。焊下 Q905 测,发现 TIP50be 极、ce 极均开路。两个结同时损坏很少见,故怀疑还有损坏的元器件存在。再查发现 12V 稳压管 D917、27V 稳压管 D916、DW3 短路或开路。换上相同参数的元器件后,该机工作正常。

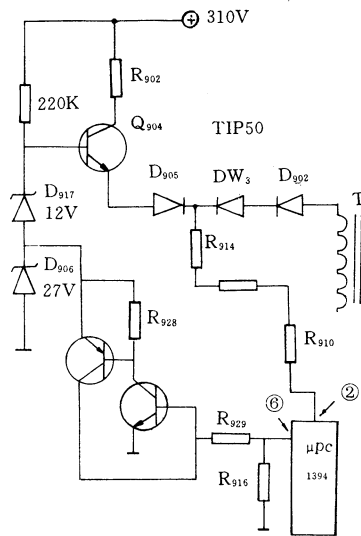


图 8

这个故障是保护电路损坏而引起,仔细查看该电源板,发现该线路元器件耐压值选择过低,如输出端 150V 档滤波电容 C927 耐压仅 160V,15V 档滤波电容 C928 耐压为 16V,很明显原设计选件不当。对整个线路功率部分的关键元器件细查,如整流桥 D901 的耐压, Q904、Q901 的耐压。对耐压偏低的元器件均选择耐压高一个档次的器件进行替换,以免发生不该发生的故障。

**六、结束语**

(下转 36 页)

# F BASIC 语言的游戏程序编程技巧

## 第六讲 游戏程序实例分析(上)

山东苍山机械化学电子工业局(277700) 于春

阅读成熟的游戏,借鉴别人的经验是提高自己编程技能的重要途径。但是当我们乍接触一个陌生的程序时,往往一时看不懂,难以确定某些语句、变量的作用。读来读去,理不出头绪。

许多朋友要求指点阅读程序的捷径。我们说,没有捷径,但有一定规律可循。本讲将以“星际大战”游戏为例,系统地介绍阅读分析游戏程序的一般方法。

### 一、先做游戏

拿到一个游戏程序,先别急于阅读,首先仔细看一遍游戏简介和使用说明(可能有的游戏没有简介说明,那么只有进行第二步)。然后再玩几遍游戏,从而对该游戏的结构、游戏过程和结局有了大体的了解。我们的电脑内固化有“星际大战”游戏的程序。调入程序运行后,屏幕上画出了繁星太空,随后显示出太空卫星杀手和太空卫星,接着询问是双人游戏,还是以电脑作对手单打。按空格键则进行双人游戏。游戏中1#操纵器控制卫星杀手,2#控制卫星,用方向键可操纵卫星向八个方向运动,按A键可发射子弹(镭射),按B键可改变卫星的运行速度,游戏目的是卫星互相击毁对方。游戏中还出现飞鱼、太空站、苍蝇、陨石、火球等干扰双方作战。若击毁卫星则得10分,击中干扰得一分,若与对手或干扰相撞则发生爆炸。每场每个游戏者有四次游戏机会(四颗卫星)。双打时若一方连毁四颗卫星,则失去游戏机会,由电脑代替这一方的游戏者继续陪另一方作战,直到另一方也失去游戏机会,则游戏结束。暂停一段时间后,重新开始。若选单人游戏则按“N”键。这时游戏者与电脑进行战斗,游戏者一方的卫星仍然只有四颗,电脑一方则有无数颗。直到游戏者失去四颗卫星后,游戏结束。在屏幕的下方有三行文字提示,第一行提示双方游戏得分,第二行提示双方所剩卫星数量和游戏水平的高低,最后一行提示游戏的最高积分。

通过玩游戏,我们基本清楚了“星际大战”是一个双打游戏,游戏目的是保存自己,躲避干扰,击毁对方。每场只有四次游戏机会。根据提示,可随时了解自己的游戏得分,所剩机会次数及游戏技能水平……。

### 二、列出程序清单

“星际大战”的源程序清单如下:

```
10' "AREA 88 V7.3"(注释行)
20 T=O=U=C=H=D=I=K=L=M=N=P=S
 =V=X=Y=MA=X0=X1=Y0=Y1=PP=Q
30 DIM D(1),V(1),F(8),SC(1),S(1)
40 L=9;MA=2;N=3;V=7;C=7;S(0)=4;S(1)=4
```

```
50 X0=70;Y0=80;X1=110;Y1=60
60 D(0)=3;V(0)=2;D(1)=7;V(1)=2;
 F(0)=3;F(1)=3;MA=MA+1;U=N;M=1
70 SP. O.;CG. 1,1;PL. "T1Y2M1V9O3R1";PAL. S 1,
 13,48,38,18
80 F. I=15 TO 18;A$=A$+SCR. (I,22);N.;H=
 VAL(A$)
90 GOS. 1070
100 V.;F. P=0 TO 200;LOC. RND(27),RND(23);P.
 CH. (205);;N.;P=0
110 '
120 I=P+6;IF F(P) THEN ON F(P) GOS. 670,700,
 680;G. 150
130 IF M=P T. GOS. 390;G. 150
140 GOS. 210
150 IF M. (I)=0 T. ER. I
160 GOS. 530
170 GOS. 800
180 O. F(8)+1 GOS. 940,1040,
190 P=1-P;G. 120
200 '
210 S=STL. (P);T=STR. (P)
220 IF T<>4 T. F(I)=1
230 IF S+T=0 T. RE.
240 IF S=0 T. 350
250 S=-1*(S=8)-3*(S=1)-5*(S=4)-7*(S=
 2)-2*(S=9)-4*(S=5)-6*(S=6)-8*(S=
 10)
260 Q=D(P)
270 IF S=Q T. 350
280 IF Q>4 T. 310
290 IF(S>Q AND S<=Q+4) T. 320
300 G. 330
310 IF(S>=Q-4 AND S<Q) T. 330
320 Q=Q+1;G. 340
330 Q=Q-1
340 PP=-1*(Q>8)-8*(Q<1);Q=-Q*(PP=Q)
 +PP;D(P)=Q
350 IF T AND 4 T. IF F(I) T. V(P)=V-V(P);F(I)=0
360 DE. M. (P)=SP. (P+8,D(P),V(P),255,1);M. P
370 IF T>7 T. DE. M. (I)=SP. (12,D(P),1,80,0,P);
 POS. I,(XP. (P)+4) MOD 256,(YP. (P)+4) MOD
 256;M. I
380 RE.
390 '
400 K=K-1;IF K>0 T. RE.
410 T=0;D=0;X=(XP. (1-P)-XP. (P))/13;Y=
```



(YP. (1-P)-YP. (P))/13  
420 IF X>0 T. D=3  
430 IF X<0 T. D=7  
440 IF Y>0 T. D=D+5  
450 IF Y<0 T. D=D+1; IF X<0 T. D=16  
460 IF X AND Y T. D=D/2  
470 S=2; O=D-D(P); IF (0<O AND O<5) OR O<-3  
T. S=1  
480 IF O=0 T. S=0  
490 IF V(1-P)<>V(P) T. T=4  
500 IF M. (I)=0 OR RND(10)=0 T. T=T+8  
510 IF X \* X+Y \* Y>200 T. RE.  
520 K=RND(L); G. 220  
530 ,  
540 D=CR. (0); IF D>0 AND D<6 T. O=0; GOS. 590  
550 D=CR. (1); IF D>1 AND D<6 T. O=1; GOS. 590  
560 D=CR. (6); IF D>0 AND D<6 T. O=6; GOS. 610  
570 D=CR. (7); IF D=0 OR (D>1) AND (D<6) T. O=  
7; GOS. 610  
580 RE.  
590 X=ABS(XP. (O)-XP. (D)); Y=ABS(YP. (O)-  
YP. (D)); IF X<10 AND Y<10 T. IF M. (O) AND  
M. (D) AND F(D)=0 AND F(O)=0 T. F(O)=1; F  
(D)=1; IF D>1 T. 650  
600 RE.  
610 X=ABS(XP. (O)-XP. (D)-4); Y=ABS(YP. (O)  
-YP. (D)-4); IF X<8 AND Y<8 T. IF MO. (O)  
AND M. (D) AND F(D)=0 T. ERA O; F(D)=1; G.  
630  
620 RE.  
630 IF D<2 T. SC(O-6)=SC(O-6)+10; RE.  
640 SC(O-6)=SC(O-6)+1+F(8) \* 4  
650 DE. M. (D)=SP. (10, RND(9), RND(3)+4, 10); M.  
D; PL. "O4C"; RE.  
660 '  
670 F(P)=2; DE. M. (P)=SP. (10, D(P), V(P), 70/V  
(P), 0); M. P. PL. "O"+HEX\$(P+3)+"GG"; S(P)  
=S(P)-1; RE.  
680 IF P=M OR STR. (P)+STI. (P) T. F(P)=0; POS.  
P, X0+P \* X1, Y0+P \* Y1  
690 RE.  
700 IF M. (P) T. RE.  
710 F(P)=3; ERA P  
720 IF SC(0)>H AND M<>0 T. H=SC(0)  
730 IF SC(1)>H AND M<>1 T. H=SC(1)  
740 IF L<1 T. L=5  
750 LOC. 0, 20; P. "SCORE", SC(0), SC(1), "LEVEL"; P.  
"SHIPS", S(0), S(1), L; " "; P. "HI-SCORE", H,  
760 IF S(P)>0 T. RE.  
770 IF M=P T. L=L-1; S(P)=4; G. 740  
780 IF M=-1 T. M=P; P. P; " COM"; PL. "EE";  
RE.  
790 LOC. 10, 10; P. "GAME OVER"; PL. "O1C5CD  
ED6C4C5"; PAU. 200; RUN  
800 '  
810 X=XP. (P)/8-1; Y=YP. (P)/8-2

820 IF X<0 OR X>27 OR Y<0 OR Y>23 T. 880  
830 IF SCR. (X, Y)<>CH. (207) T. 880  
840 S=ASC(SCR. (X, Y, 1))  
850 IF S=P+1 T. 880  
860 IF F(P)=0 T. F(P)=1  
870 '  
880 IF M. (I)=0 T. RE.  
890 X=(XP. (I)-12)/8; Y=(YP. (I)-20)/8  
900 IF X<0 OR X>27 OR Y<0 OR Y>23 T. RE.  
910 IF SCR. (X, Y)<>CH. (207) T. RE.  
920 S=ASC(SCR. (X, Y, 1)); IF S=P+1 T. RE.  
930 ERA I; LOC. X, Y; P. " "; ; RE.  
940 '  
950 F. I=2 TO MA; IF M. (I) OR F(I)=2 T. 980  
960 IF F(I)=1 T. GOS. 990; G. 980  
970 DE. M. (I)=SP. (C, RND(9), RND(3)+C/2, 80, 0);  
M. I  
980 N. ; RE.  
990 '  
1000 F(I)=0; ERA I; POS. I, 120, 120; IF U<MA T. F(I)  
=2  
1010 U=U-1; IF U=0 T. F(2)=0; F(8)=1; RE.  
1020 IF U<0 T. U=N; F(2)=0; F(3)=0; F(4)=0; F(5)  
=0; F(8)=0; C=C-2; L=L-1; IF C<2 T. C=7;  
IF M>-1 T. S(1-M)=S(1-M)+1  
1030 RE.  
1040 IF F(2) T. I=2; G. 990  
1050 IF M. (2)=0 T. DE. M. (2)=SP. (2, RN. (8)+1,  
RN. (3)+1, 10, 0); M. 2  
1060 RE.  
1070 '  
1080 V. ; RES. 1070  
1090 F. P=0 TO 200; LOC. RN. (27), RN. (23); P. CH.  
(205); N.  
1100 LOC. 8, 3; P. "STAR KILLER"  
1110 F. P=0 TO 1; LOC. 10, 10+P \* 8; DE. M. (P)=SP.  
(P+8, D(P), 2, 30); POS. P, X0+P \* X1, Y0+P \*  
Y1; M. P  
1120 IF M. (P) T. 1120  
1130 READ A\$: F. I=1 TO LEN(A\$); P. MI. (A\$, I,  
1); ; PL. "C"; N. ; N.  
1140 F. P=2 TO 3; LOC. 10, 17+P; READ A\$: F. I=1  
TO LEN(A\$); P. MI. (A\$, I, 1); PL. "C"; N. ; N. ;  
LOC. 1, 22; P. "2 PLAYER?"(Y/N); ; M=-1  
1150 A\$=INK. (0); IF A\$="N" T. M=1  
1160 PAU. 80; RE.  
1170 D. "PLAYER 1", "PLAYER 2", " OR",  
"COMPUTER"

源程序中的 110. 200. 390 等行为注释行。由于注释使用日文故略去。另外 830、910 行中的 CH. (207)，在源程序中为背景图案，为便于印刷而改动。

### 三、程序分段

阅读程序，一般遵循先整体后局部、先易后难的原则。首先从整体上把程序分段。模块结构程序可根据 GOSUB 语句的行号划分功能模块和子程序；顺序结

构的程序一般按 GOTO 语句的行号分段。“星际大战”共有 13 条 GOSUB 语句,行号分别为 210、390、530、590、610、670、680、700、800、940、990、1040、1070。这些行号一般为功能模块或子程序的入口行号。然后再根据 RETURN 语句确定模块或子程序的结束行号,从而可分出 11 个模块,分别为:

- |               |               |
|---------------|---------------|
| 1. 240~380    | 2. 390~520    |
| 3. 530~580    | 4. 590~670    |
| 5. 680~690    | 6. 700~790    |
| 7. 800~930    | 8. 940~980    |
| 9. 990~1030   | 10. 1040~1060 |
| 11. 1070~1170 |               |

不难看出 20~80 行为初始化赋值程序,我们也可把它列为一个模块,称为模块 0,行号为 20~80。这样整个程序共划分 12 个模块。

#### 四、写出管理程序

程序分段后,剔除以上 12 个模块,就可容易地列出管理程序。程序清单如下:

```
10 REM "STARS KILLER"
20~80
90 GOS. 1070
100 V. : F. P=0 TO 200: LOC. RN. (27), RN. (23): P.
 CH. (205): N. : P=0
110 REM "MAIN"(主程序)
120 I=P+6: IF F(P) T. O. F(P) GOS. 670, 770, 680: G.
 150
130 IF M=P T. GOS. 390: G. 150
140 GOS. 210
150 IF M. (I)=0 T. ERA I
160 GOS. 530
170 GOS. 800
180 ON F(8)+1 GOS. 940, 1040
190 P=1-P: G. 120
```

现在清晰多了。从主程序可见,游戏的进行过程就是在 120~190 行之间反复运行。

#### 五、大体确定各模块的功能

写出管理程序后就可分别分析,逐个确定模块功能。对于一时较难确定功能的模块,可暂时跳过,待其

它模块功能确定后,在详细分析程序时再对其进行专题研究。

现在我们进行这项工作:

1. 模块 0: 20~80 行。为初始化模块。
2. 模块 1: 210~380 行。读操纵器,启动 8<sup>#</sup>、9<sup>#</sup>、12<sup>#</sup>卡通运动。
3. 模块 2: 390~520 行。这段程序功能不太明确,暂往后放。
4. 模块 3: 530~580 行。为测试各动作间是否发生碰撞程序。
5. 模块 4: 590~670 行。测发生碰撞的两卡通之间的距离。若距离较近则启动爆炸动作。爆炸为 10<sup>#</sup>卡通。
6. 模块 5: 680~690 行为确定 P<sup>#</sup>动作的起始座标子程序
7. 模块 6: 700~790 行的功能是打印屏幕提示和结束处理。
8. 模块 7: 800~930 行。这段程序不好理解,且对整个程序的运行影响不大,可以先删去。删去的理由是屏幕上没有 207 号图案,因而程序执行时必然从 830 行转 880 行,再返回主程序,而且开始的 810 行计算的 X、Y 值,主程序也不使用。因此,可暂删去 170 行。
9. 模块 8: 940~980 行。定义 C<sup>#</sup>号卡通。从 40 行可知 C 的初值为 7, 7<sup>#</sup>卡通是太空站。其中 960 行还要调用 990(即模块 9),而模块 9 仅为 960 行所调用,所以可把两模块合并。其功能是定义干扰物子程序。从 1020 行可知 C 等于 7、5、3。因此,对应的干扰卡通分别为太空站、飞鱼、火球。
10. 模块 10: 1040~1060 行。定义 2<sup>#</sup>卡通(苍蝇)运动。
11. 模块 11: 1070~1170 行。为标题和选项处理程序。

通过以上工作,除模块 2、7 的功能暂不明确外,其它模块的功能已大体确定。

(待续)

## 家用电子游戏机故障维修一例

烟台计量仪表厂(264000) 杨 林

红白“小灵通二代”电子游戏机,故障现象为 START 键失效,游戏不能开始,或按某一键偶尔有效,但结果却进入自动循环过程。

经初步查操纵盒 I 连线,发现连线有断路现象,连接好后继续试验,故障依旧,故怀疑问题出在主机板

上。操纵盒与 CPU 连线如图 1:

CPU 的 39 脚 LOAD 加载信号,供给 4021 作为串出并入的控制端,31 脚 RDY 信号为脉冲信号,经 74HC368 驱动后,作为 4021 的时钟信号。4021 的原理框图如图 2 所示。

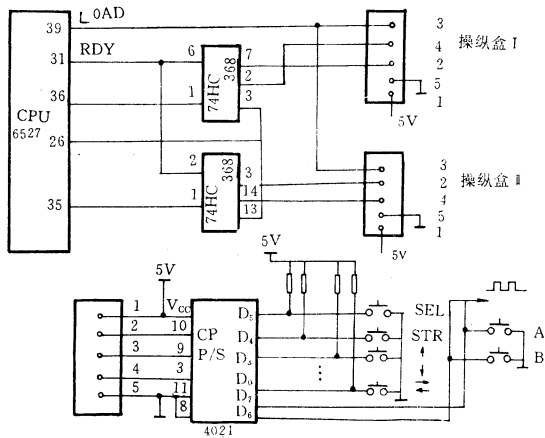


图 1

当 P/S 为高电平时,封锁 CP 时钟信号,数据由  $D_0 \sim D_7$  并行输入 D 触发器的输入端;当 P/S 由高电平变为低电平时,CP 时钟信号有效,在 CP 下降沿将并行输入的数据向右串行移一位。P/S 由 6527 的 Load 信号提供,CP 由 6527 的 RDY 信号提供。由于 LOAD 信号的高低电平时间都为 10ms,而 RDY 信号的周期为 0.6 $\mu$ s,因此保证了并行输入的数据在一个 LOAD 周期内完全由  $Q_n$  输出所属的 CP 时钟个数,亦就是说,

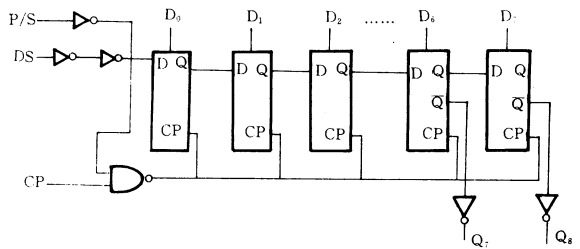


图 2

CPU 发出一个 Load 信号,便由  $D_0$  端串行输入 AB 键,START SELECT 键,上下左右键状态给 CPU 判断为前按键状态。6527 的 35、36 脚的 1NP0、1NP1 信号,为读操纵盒 I 和 II 的选通信号。由以上分析可知,只要 LOAD、RDY、1NP0、1NP1 信号有故障,都将导致操纵盒失控。

经查 CPU39 脚电压不为正常的 0.07V,换一芯片后,故障排除。

## 征集电子图书书稿

电子工业出版社是国家级专业出版社,具有很强的编辑力量和较高的声誉。为了推动我国电子科学技术的发展,解决有识之士“出书难”的问题,同时满足各位涉足电子领域人士的求知需要,我们采取这种公开征稿的方式。欢迎各位热心作者及单位踊跃投稿,也欢迎热心读者点题,以建立我们与作者、读者之间的友谊热线。

### 一、征稿题目

- (1)最新黑白电视机原理、使用与维修
- (2)最新组合音响原理、使用与维修
- (3)最新袖珍收录机(随身听)原理、使用与维修
- (4)最新激光放像/放唱机原理、使用与维修
- (5)最新摄录一体机原理、使用与维修
- (6)最新游戏机原理、使用与维修
- (7)最新卡拉 OK 机原理、使用与维修
- (8)最新电冰箱原理、使用与维修
- (9)最新洗衣机原理、使用与维修
- (10)最新复印机原理、使用与维修

- (11)最新空调器原理、使用与维修
- (12)最新电脑笔记本原理、使用与维修
- (13)最新电子打字机原理、使用与维修
- (14)最新传真机/电传机原理、使用与维修
- (15)最新家用保健器具原理、使用与维修
- (16)最新家用电器(微波炉、电熨斗等)原理、使用与维修
- (17)最新计算机音乐原理与实践
- (18)最新电子、电动玩具的原理,使用与维修
- (19)最新电子琴原理、使用与维修

### 二、编写要求

每个题目所涉及的技术最新;机型最新;原理最详细;使用最合理;维修最简洁。

### 三、投稿要求

每位投稿者都要提供自己的简历,书稿的编写大纲;编写大纲的自我评价及编写计划。经我社审批后,即可列选。

### 四、投稿方式

每位投稿者将所要求的材料直接邮给:北京电子工业出版社第四编室,邮编:100036;电话:8212233 转

3419 或 3418。也可直接到出版社联系和询问有关事宜。本社地址:北京市万寿路 27 号院。

## 卫星电视高科技的享受

卫星电视设备不仅可用于城市机关、团体、宾馆、饭店、家属区闭路电视和个人家庭使用,而且特别适用于油田、矿山、地质、林场、农场等地处偏远或流动作业点不能收看电视或接收效果差的单位和个人使用。

卫星电视设备可直接接收由卫星转播下来的中央一、二、四台,云南、贵州台等国内电视节目和亚洲一号卫星转发的香港五套电视节目(体育台、中文台、音乐台、合家欢台、BBC 新闻台,这五套节目二十四小时全天播映),比录像机更为经济实惠,可为你处职工活跃文化生活,扩大知识视野,提供经济信息。

本公司专业经销 1.5m、2m、2.4m、3.2m、4.5m、6m 卫星电视接收系统及配套设备,其中 3.2m 系统适  
通信地址:北京市万寿路 173 信箱

(万寿路北口)

邮政编码:100036

联系电话:8221677 815342

用于宾馆、饭店、住宅楼闭路电视及林区、矿山、油田电视差转台使用;2m、2.4m 系统适用于边防哨所、检查站等小集体和常年从事野外作业的流动性团体使用(可配太阳能电源);1.5m 适用于个人家庭。

本公司以最低价格经营服务并负责安装、调试、维修及技术咨询,图象质量达到四级标准(五级为广播级)。卫星闭路电视系统酌收安装费。本公司 1993 年将根据用户的要求举办亚星技术培训班,定期为用户邮亚星节目单。广播电视在发展,新技术在不断推广,新的一年本公司将以精益求精之精神与各位朋友合作,开拓推广市场。

开户银行:北京工商行翠微路分理处

户名:北京富益电子技术开发公司

帐号:046054-07

联系人:徐轩

## 台山市化工厂向您提供 优质碳酸钡

“台化牌”碳酸钡纯度高,质量稳定,含铁、钠等杂质极微,已广泛应用于电子、玻璃、磁性材料、无性电元件、陶瓷、显像管玻壳等工业方面,深得广大用户信赖。本厂竭诚欢迎新老用户广泛合作。

欢迎广大用户来人来函洽谈订货

厂址:广东台山市城南门路

电话:07657-524901

电挂:3602

邮编:529200

联系人:甄国超

## 敬告读者

原本编辑部设在北京万寿路 27 号北京振兴电子公司的《电子与电脑》读者服务部已撤销,诸有关事宜请与本刊编辑部联系办理。

## 邮购消息

最新裕兴游戏机家庭电脑,集学习、游戏功能于一体,具有浮点 BASIC;F BASIC;声像学习;中文输入、编辑打字;电视游戏等多种功能。超级学习卡创新设计,用户无须插卡即可扩充各种学习和游戏软件。邮购:388 元/套(含键盘、超级学习卡,操作手册,打印机接口,不含游戏机)。邮费:12 元。保修一年。地址:北京西城新街口西教场 35 号(100035)。电话:2252309。2241471。备有资料,欢迎垂询。来信请附回信信封,帖足邮资。

# RS232/TTL电平转换芯片MAX232的

## 原理及其应用

罗维国 吴微

MAX232系日本MAXIM公司为串行口RS232生产的专用驱动芯片,同类产品有TC232等。在单5V供电条件下,能将0/5V(TTL电平)转换为+10V/-10V(RS232电平),将+10/-10V(RS232电平)转换为0/5V(TTL电平)。能直接替代传统的接口芯片MC1488和MC1489,并省去±12V电源,使用极为方便。下面对其工作原理进行分析,并提出几点应用。

### 一、MAX232的工作原理

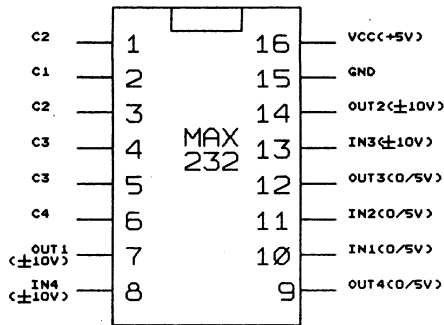


图1 MAX232引脚图

MAX232引脚图如图1所示,1~6脚为电容端子,7~14脚包含4路RS232/TTL电平转换器,其中两路为0/5V→+10/-10V,另两路为+10/-10V→0/5V。对于MAX232来说,首先它是一个电平转换器,采用电荷泵电压变换方式,下面可由+5V转换为±10V为例说明电压变换原理,如图2所示,C1~C4为外接电容,RL+和RL-为外部负载。MAX232内部有一振荡器,工作频率为25kHz,S1~S8为片内模拟开关。由+5V变换为+10V的原理如图3所示。假设C3上起始电荷为零,在 $t_0$ 时刻,S1、S3闭合,+5V对C1充电,如图3.1所示。下一时刻 $t_1$ ,C1对C3充电,如图3.2所示。再下一时刻 $t_2$ ,+5V对C1充电,同时C3上的电荷与+5V对负载RL+共同供电,如图3.3的所示,于是+10V电压得以产生。至于-10V的产生,则是在+10V的基础上产生的。参照图4,按照上述分析方法,读者不难看出其电平变换原理。

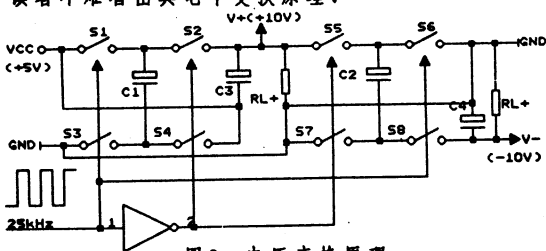


图2 电压变换原理

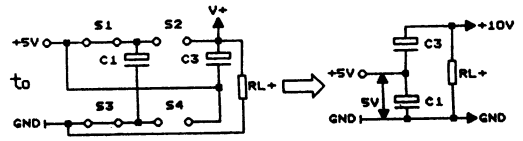


图3.1 C1充电

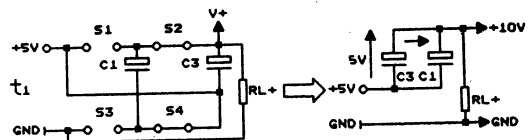


图3.2 C1对C3充电

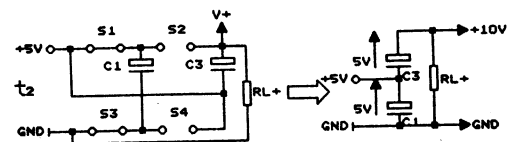


图3.3 C1充电、C3供电

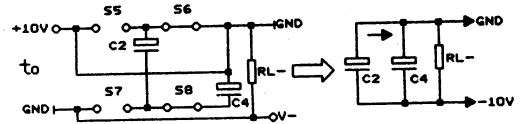


图4.1 C2对C4充电作

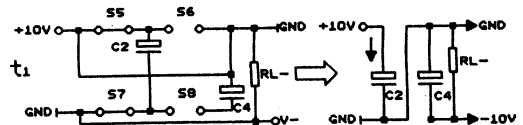


图4.2 +10V对C2充电

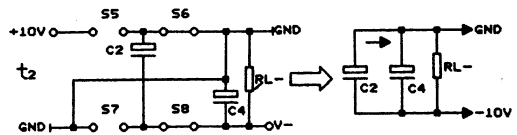


图4.3 C2对C4充电

对于OUT1(7脚)来说,在VCC加上后,可以输出+10V,亦可输出-10V,其电压值由控制端IN1(10脚)上的电平决定。鉴于RS232电平的负逻辑约定,10脚的0V对应7脚的+10V,而10脚的5V对应7脚的-10V。

至于由±10V转换0/5V,采用类似原理完成,不必赘述。



## 二、MAX232的应用

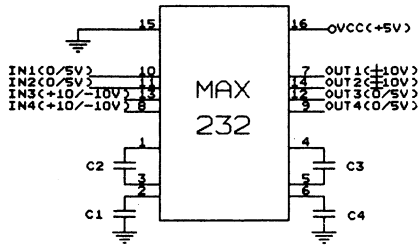


图5 MAX232基本应用

MAX232芯片的典型用法如图5所示，C1~C4取10 $\mu$ F即可，下面介绍几个应用实例。

### 1. RS232/TTL电平转换接口

目前广泛使用的单片机8031、8098的串行口TXD/RXD与RS232接口时，传统上采用MC1488和MC1489，需外加 $\pm 12V$ 电源，另一种方法采用分立元件构成电平转换电路，电路复杂，可靠性差。由MAX232完成RS232/TTL电平转换，勿需 $\pm 12V$ 电源，硬件简单，工作可靠。应用实例如图6所示。

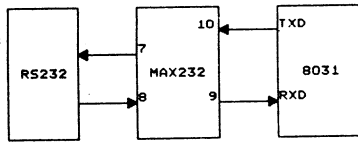


图6 MAX232用于通讯接口

### 2. $\pm 10V$ 电压发生器

在不少场合，往往需要正负对称的电源，如双电源集成运级，或绝对值大于5V的正负电源，如A/D、D/A的参考电压。单独为此设置外接电源势必增加硬件开销，而有些电源变换芯片（如：TL497、TL499等芯片）只有一路输出。MAX232的 $\pm 10V$ 输出在负载是电阻为10K时输出电压不低于9.0V，负载是电阻为3.9K时不低于8.5V，具有一定的带载能力。典型用法如图7所示。

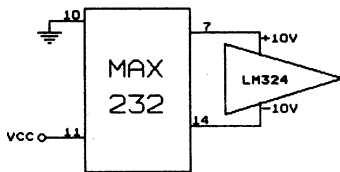


图7  $\pm 10V$ 发生器

### 3. 频率发生器

MAX232的片内振荡器频率25kHz，该振荡器勿需外接元件，为真正独立的片内振荡器，工作稳定可靠。对于某些需要频率发生的场合，尤其是将MAX232作通讯接口或正负电源发生器的情况下，可从第3脚引出振荡信号（TTL电平），这样能

做到一片多用，物尽其用。

为了方便读者用好MAX232，请向下述邮购。  
每片价格25.00元(含邮费)。

武汉市武昌珞瑜路100-1号 武汉创意电子研究所  
邮编：430072 电话：712761  
联系人：宋咏良

广州市天河科技街华南师大科技服务楼 215号  
电子工业出版社广州科技公司  
邮编：510631 电话：  
联系人：王惠民

## 单片功率开关电源 IC

广东工学院 李秀华 吴剑棠

随着IC技术的发展，开关电源不但能够把控制部分做成通用性高的集成电路，而且能把功率部分也制造在集成块内，形成所谓单片式功率开关电源。表一列出了一些单片功率开关电源IC的型号及其主要性能特点。下面我们将以美国POWER公司生产的PWR-SMP3为例说明单片功率开关电源的内部结构原理，最后给出PWR-SMP3的一个应用实例。

PWR-SMP3集成块有三个显著的特点，一是它采用高反压的金属氧化物场效应晶体管(MOSFET)，能够在高电平运行，芯片可直接接到120伏的电源线表一：单片功率开关电源

| 型号             | 输入电压<br>V <sub>imax</sub> (V) | 振荡频率<br>(KHz) | 输出电流<br>I <sub>omax</sub> (A) | 封装型式   |
|----------------|-------------------------------|---------------|-------------------------------|--------|
| L4962          | 50                            | 200           | 1.5                           | DIP-16 |
| L4964          | 50                            | 200           | 4                             | 单列15线  |
| L4974          | 50                            | 500           | 3.5                           | DIP-20 |
| L4970          | 50                            | 500           | 10                            | 单列15线  |
| PWR-SMP3       | DC200V<br>AC130V              | 1MHz          | 1                             | DIP-16 |
| PWR-SMP<br>260 | DC200V<br>AC130V              | 400           | 2                             | 单列     |

|                |                  |     |     |            |
|----------------|------------------|-----|-----|------------|
| PWR-SMP<br>400 | 72               | 750 | 1   | DIP-16     |
| PWR-SMP<br>110 | DC200V<br>AC130V | 850 | 2   | DIP-16     |
| PWR-SMP<br>120 | DC200V<br>AC130  | 330 | 4   | DIP-16     |
| MC34063        | 40               | 100 | 1.5 | DIP-8      |
| MC34166        | 60               | 100 | 3   | T0-220(5线) |
| LM2575         | 35               | 52  | 1   | T0-220(5线) |

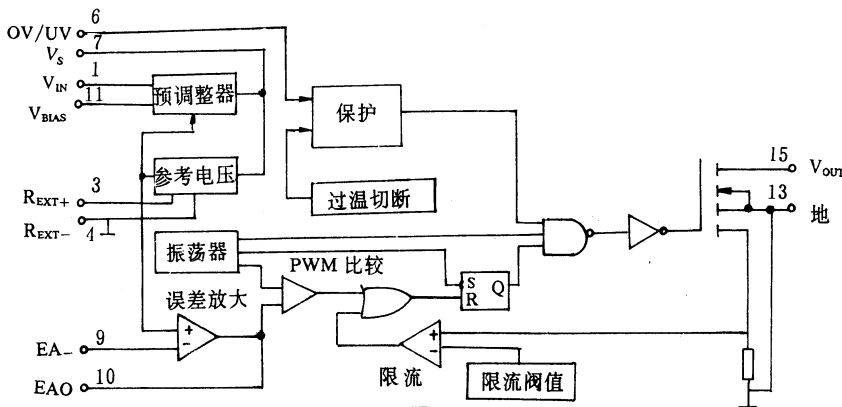


图 1

上,而一般其它芯片只能接低电压输入。并且其高速开关特性比双极性晶体管好;第二个特点是其输入电容和米勒电容很小,从而使门驱动容易并能提高开关速度,其工作频率达 1MHz,大大减小外接储能元件的体积;第三个特点是芯片的逻辑电平阈值电压只需 5V,而一般器件的电平值为 10V。这样既可减少驱动功率,又可直接与 COMS 模拟及数字电路(如运算放大器、比较器、逻辑门、寄存器等)连接,简化了中间连接的元件数,使可靠性大大提高。此外,它还具有快速电流限制和快速温度保护等特点。

PWR-SMP3 的内部框图如图 1 所示,下面将其各个主要部分逐一介绍。

### 1. 预调整器

预调整器是当电源开始接通时给控制电路和驱动电路提供初始偏置电流,它由高压 MOSFET 管,门偏置电流源和误差放大器组成,其中误差放大器通过控制 MOSFET 管的栅极而把  $V_s$  调节到大约 5.6V 左右, $V_s$  的引出端 7 脚外接旁路电容给门驱动开关电流一个通道从而减少电流噪声。11 脚是接由电源输出反馈绕组引来的偏置电压  $V_{BIAS}$ ,其典型值为 8.5V,以便在电源正常工作时关断预调整器,这样可提高效率,减少损耗。

### 2. 参考电压 $V_{REF}$

$V_{REF}$ 用来设定误差放大器、过温保护电路、限流电路的阈值的。

### 3. 振荡器

振荡器是一个自激振荡器,内部电容通过开关恒定电流源交替地充电和放电,开关电压由比较器决定,周期由电流源的上升和下降的斜率决定,是一个固定值,频率为 1MHz。

### 4. 误差放大器

误差放大器主要是由一个高性能的运算放大器组成,误差放大器的输出级是一个缓冲器,其偏压为 2V,输出阻抗为 1.5K $\Omega$ ,误差放大器的输出直接控制功率开关的工作周期。

### 5. 脉宽调制器

电压型的脉宽调制器把由误差放大器输出的控制电压和由振荡器产生的锯齿波电压进行比较,产生的驱动信号控制功率开关,驱动信号的周期随输入电压和负载的变化而改变,增大周期令输出电压升高,反之,减小周期则令输出电压下降,从而使输出电压保持稳定。

### 6. 过压/欠压关断保护

当输入电压过高或过低时,该保护电路会使功率开关关断,

OV/UV 输出端(引出脚 6)外接一个电阻分压器,它决定关断的电压值。

### 7. 限流保护

限流检测由一个镜式检流计和一个感应电阻组成。检流计产生一个与功率开关管漏极电流成比例的电流,再通过感应电阻产生感应电压,这个电压通过比较器与参考电压进行比较,决定是否需要限流。

### 8. 过温保护

当功率开关的结温过高时(典型值为 135 $^{\circ}\text{C}$ ),温度保护电路将会切断功率开关,其响应速度比一般器件快,当温度降到安全水平时,又重新接通功率开关。

图 2 是应用 PWR-SMP3 构成离线开关电源的一个例子,它采用反馈式(或称回扫式)结构,其工作原理已在上期作过介绍。这种方式对 100W 以下的小功率电源较适用。此例当采用 T1001 标准变压器,输入 85~140V 交流电压时,其输出电压为 5V,输出功率为 5W。

影响输出电压稳定的三个主要因素是:反馈绕组电压是否稳定;功率变压器的耦合是否良好以及怎样使用脉冲变压器来消除电压尖峰。图中  $L_2$ 、 $L_3$ 、 $L_4$ 、 $C_6$ 、 $C_7$ 、 $C_8$ 、 $C_{12}$  和  $C_{14}$  组成抗电磁干扰滤波器, $D_1$ 、 $D_6$ 、 $D_7$ 、 $D_8$  和  $C_1$  把交流输入电压整流成直流电压, $R_1$  和  $R_2$  组成分压器决定了欠压/过压保护的关断电压值, $D_5$ 、 $C_9$  和  $R_7$  组成电压钳位电路以限制开关管漏极上的峰

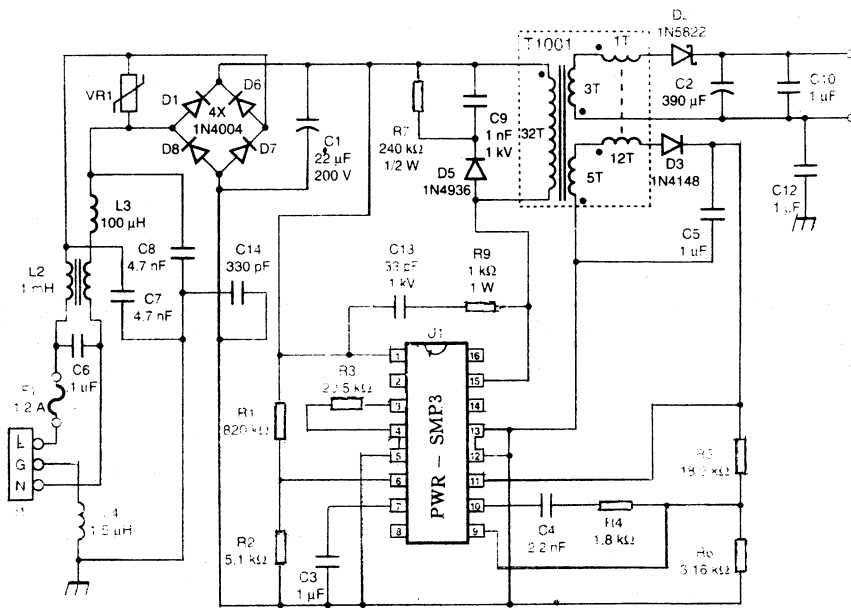


图 2

值电压,  $C_{13}$  和  $R_9$  抑制由漏感引起的振铃电压,  $R_5$  和  $R_6$  确定了反馈电压为 8.5V,  $R_3$  设定了 PWR-SMP3 内的电流源,  $C_3$  和  $C_5$  是旁路电容。  $D_2$  应选用正向电压小而逆恢复特性好的肖特基整流管以减少二极管的开关损耗。该电源在满负载及正常工作电压条件下效率为 65%。

## 求 援

(上接 56 页)

此法虽简,但肯定要失败,理由有二:

(1)DOS 系统盘是配套使用的,换句话说,要教治不可自举的硬盘,并非任一“正常 DOS 系统盘”均可胜任;而是应找与该硬盘所装的 DOS 生产厂家相同的系统软盘来用(因为硬盘的 BOS 引导记录(不是主引导记录!)只认其相应的两个系统隐含文件);且最好是用随机配发的 #1 盘。

(2)即便如此,仅按《王》文的简便方法也仍是不行的。因为大家知道,软盘与硬盘的 BOS 引导记录区中的 BPB 参数是大不相同的。具体些说,诸如:扇/簇,文件数,总扇数,介质说明(FD, F8, F9, ……), FAT 长度,扇/道,头(面)数,隐扇区数,等等,软盘的与硬盘的是相距甚远的,这些差异在 BPB 中均已明示,或者说,在 BOS 引导主录区自开头偏移 0B—1D 处,二者是不同的,也是不可互换的,所以仅按《王》文方法处理,硬盘的自举是不可能的;而且,即使经软盘进入硬盘,连 DIR 都不会正常,别的更不必说了。

正确的作法是:依《王》文所说处理后,再参照有关资料(因硬盘容量、DOS 版本不同,这些参数稍有不同),用 DEBUG 将 0B—1D 这不足 20 个字节中的有关者予以修改,而后方可达到目的。

本人有一台深圳华明计算机厂出产的 CEC—IG 型机,不慎把电源部分一个叫“M3131”的集成块(起振荡作用)烧了,曾多次到各处购买,均未能找到,不知何处可买到此种型号的集成块或其代用品,或中华机电源。

张晓农

联系地址:湖南城步城南路 49 号 邮政编码: 422500

△ △ △

·我有一台中华学习机出了故障,开机喇叭正常,可能是  $U_{17}$ N4008 集成块损坏,特求购以下集成块:  $U_1$ 6502、 $U_3$ 50464 (64 × 4B 动态 RAM)  $U_{17}$ N4008、 $U_4$ N4006、 $U_{15}$ N4007、 $U_{26}$ 2716 (24K 字节即 ROM)  $U_{13}$ 2732 (4K 字节 EPROM)、 $U_{30}$ TCA650、 $U_{40}$ (PROM256 × 8)  $U_{40}$ 中华学习机书中没有注明此集成块型号、 $U_{25}$ KB3600、 $U_7$ 27256。

福建省天湖山矿务局汽车队 胡校  
邮 362617

(上接第 2 页)

订。实现信息共享和各类指标规范,如各类指标设置、存储分类、项目编码、信息交换规格、资料格式、技术手段、工作程序、信息传播介质等。

5. 尽快制定适合我国用户的数据通信技术业务管理规程和资费标准,加强数据通信宣传,不断促进业务发展。

# 应用程序接口 (API) 时代的开端

文 海

网络管理者已进入了使用应用程序接口的时代。API 可帮助用户在新的机器、新的环境以及看起来是不可沟通的不同通信协议之间安装应用程序,因而 API 对现在流行的获取网络交互操作能力的方法提出了挑战。

API 是工作在特定环境下的应用程序员为了完成某一任务所使用的子例程库。这些子例程能使程序员在对实体的内部工作情况一无所知或知之甚少的情况下,直接使用象驱动器、打印机、文件系统和网络等基本实体所提供的服务。

每个面向最终用户的应用程序员都可通过 API 来调用专业通信程序员们的详细工作。举例来说,为了传输数据,应用程序员无需知道传输协议的头格式或错误情况,他们只需将要传输的数据及传输的目的地放入缓冲区,键入“发送”,数据就会通过网络传输。

API 甚至可以分成层,把不同类型和复杂程度的 API 隐藏在每一级。

API 至少可同时完成两个功能:一、它们可使为一种协议所编的应用程序不需要或只需要很少的改动就可在另一种协议上运行。二、它们可使复杂的基本网络实体看起来很简单,且易于为应用程序员所使用。

API 的发展对网络工作者来说是一个重要的发展。当今,大多数的计算机应用程序与它们的系统环境(包括操作系统以及程序语言、数据库和网络协议等子系统)是硬连线的。这意味着这些应用程序是不可移植的。例如,将运行在 IBM 的 SNA/LU6.2 下的,用 COBOL 语言编写的 IBM CICS 联机事务处理系统移植到使用传输控制协议/网间协议(TCP/IP)或开放系统互连(OSI)的 Unix 平台上是大多数信息系统管理员所不愿考虑的,除非这种移植能带来好处。

网络 API 的特点是它们应用程序独立于网络结构。具有交互操作能力的关键在于使用了 API,而不是协议。

八十年代,网络管理者、供应商及其顾问们花费了大量时间来讨论哪一种网络协议是最有效、最强大,同时又最简单、最正确?所有网络,不管是广域网还是局域网,是否必须使用经过测试验证了的传输控制协议/网间协议(TCP/IP)?更进一步,客户服务器应用程序是建立在 Unix 远程过程调用(RPC)机制上,还是建立在 IBM 的 LU6.2 以及与其相关的高级程序到程序通信接口?是安装 OSI 的会话层好,还是等待 OSI 的事务处理协议?关于电子邮件、文件传输协议甚至操作系

统本身也有争论发生:是否使用 Unix?如果使用,使用哪一种?是否使用其它的操作系统?使用哪一种?

进入九十年代,讨论向更高的也更实际的一层发展:用户正在选择能最好的提供基本服务实体的程序接口来代替协议。

如果应用程序员是通过标准高级 API 来获取通信功能,那么具体是使用 OSI、TCP/IP 还是 SNA 会有什么影响呢?如果他们通过 API 来获取电子邮件服务器和开关函数。那么 X.400 作为标准信息的不可动摇的地位会不会受到威胁呢?

API 的崛起改变了关于交互操作能力的讨论。

对 MIS 系统管理员来说最急迫的问题是,传输 API 是否能从应用程序的角度将 TCP/IP、SNA 和 OSI 的区别完全掩盖掉;更进一步,能否将 X.400 邮件服务器、CICS 事务程序以及 RPC 客户服务器的区别掩盖掉。如果 API 是 AT&T 开发的、基于 Unix 的传输层接口,答案是几乎可以掩盖掉。传输层接口(TLI)提供了象 Internet 的 TCP 和 OSI 的 TD4 那样的公共程序接口,但是它没有完全掩盖掉两种协议之间语义上的区别,所以程序员还必须知道他们所使用的是哪一种协议以及如何满足一些特殊的需要。实际上,在传输层接口(TLI)之上还需要一个非标准的适配层来消除语义区别。

IBM 最近宣传 CPI-C(一种最初作为 LU6.2 的接口的专有传输 API)作为 SNA、TCP/IP 和 OSI 的公共接口,其意图是减弱 TCP/IP 和 OSI 对 SNA 的竞争威胁。IBM 已向位于英国的 X/Open 公司注册了阅读 CPI-C 的许可,来推动其他厂商把这种接口应用到非 IBM 操作系统和传输协议。X/Open 是一个推动开放系统发展的国际性组织。

虽然理论上的东西还有待实际工作中软件的检验,但为了完全掩盖掉传输提供者,我们必须依靠 API。

实际应用中,还需将 API 分层。例如,象远程过程调用这样的高级 API 或象 Covia 公司的通信集成器这样的信息 API 能提供真正的传输独立于网络的应用程序,只要适当降低层次,一个中间程序就可处理所有的工作。这些中间程序,即所谓的“中间软件”,可使用一种或多种 API 来获得基本传输服务。

低级传输 API 并不是真正透明的,高级 RPC 或信息 API 几乎是透明的。但这些 API 的工作情况都有性能或价格上的缺陷。

# 传真机原理(下)——调制和解调

张建军 孙晓春 张景生

传真图象信号在编码后是数字信号,这些数字信号是不能直接在现有的模拟信道上传输的。这是因为:(1)目前已有的模拟信道大都是带通型(每个电话电路的频带宽度为 300~3400Hz),而在传真数据信号中,有很多的低于 300Hz 和高于 3400Hz 的成分不能通过现有的电话电路,如果直接传输会造成严重失真。(2)数字信号抗干扰的能力不强。(3)传真数据信号占有很宽的频带,直接传输不利于有效地利用信道带宽,也不便于构成信道的复用和与目前大量使用的模拟信道兼容。为了解决这个问题,对传真信号采用了调制的方式。

在三类传真机,调制解调器的作用就是将传真图象信号进行调制和解调。

CCITT 在关于三类传真机的建议中,对三类传真机所采用的调制解调器作了相应的规定:在公用电话交换网上使用的三类传真机应采用建议 V. 27ter 中规定的调制、扰频、均衡和定时信号;在租用电路或高质量的交换电路上使用的三类传真机,采用建议 V. 29 中规定的调制、扰频、均衡及定时信号;三类传真机传输控制规程的传递,则采用建议 V. 21 第二信道的调制方式。

## 一 G3 调制解调器

主要完成以下功能:

- 1)、发送时,在阶段 C,产生单音信号,调制 300 波特的控制信号。
- 2)、接收时,在阶段 C,同步检测接收信号、滤波等。
- 3)、接收信号的自动均衡、相位调整。

将复杂的网络隐藏于象“发送”和“接收”这样的简单的命令集对那些使用视觉 BASIC 而不是使用 Windows 的软件开发工具开发程序的人有很大好处。尽管后者在技术上更有效,但使用视觉 BASIC 的人大大多于使用 SDK 的人。

虽然高级网络 API 需要很多处理周期和带宽来使应用程序在网络上工作,但它们将使开发者开发出比今天的终端用户或局域网上的应用程序更好的应用程序成为可能。

当一个广泛使用的 API 过时后,事情会变得很糟糕。例如,DOS 实际上是 16 位 IBM 兼容机的 API,当 32 位处理器出现后,DOS 限制了内存的使用。但现有

4)、传真信号的扰频。

G3 调制解调器采用的调制方式,符合 CCITT V 系列有关建议,其主要性能如下:

1. 调制方式:

- 9600bps 时采用八相位、4 幅值正交调幅 (CCITT V. 29 建议);
- 7200bps 时采用八相位、2 幅值正交调幅 (CCITT V. 29 建议);
- 4800bps 时采用八相编码调制 (CCITT V. 27ter 建议);
- 2400bps 时采用四相差动编码调制 (CCITT V. 27ter 建议);

2. 调制速率:

- 9600bps 时为 2400 波特±0.01%
- 7200bps 时为 2400 波特±0.01%;
- 4800bps 时为 1600 波特±0.01%;
- 2400bps 时为 1200 波特±0.01%;

3. 载频

- 9600bps、7200bps 时为 1700Hz±1Hz;
- 4800bps、2400bps 时为 1800Hz±1Hz;

4. 需要线路特性(请根据线路状况调整调制解调器板的均衡器)

- 载波段 0~7 段
- 负载电缆(直径 0.5)非线性 8dB 以上
- 负载电缆长度 15 公里以上

5. 通信电平(在下列范围内不需调整)

- 发送电平:0±2dBm
- 接收电平:0~-40dBm

的 API 和服务提供实体至少提供了它们演变的方向。过时的 API 可以接到服务提供者上,使其与新的应用程序共存。在这方面,API 提供了真正的“当你不需要它就改变它”的模式,这一模式正是国际标准化组织的七层开放系统互连所承诺的。

有些人可能会对 API 虽然是开放的和公布的,但仍是专有的机制提出疑问。标准化机构正专注于他们自己的 API,但工业软件供应商们已经进入了市场,他们带来了真正的产品,对标准制定组织形成压力。API 的崛起是一个信号,长期受脱离于市场的标准制定者所影响的网络市场,将被对应用软件的需求所推动。



6. 传输时序: 仅有外接方式(ST2)
7. 自动均衡器: 在接收方, 接有适当的 MS 自动均衡器
8. 扰频器、解扰器: 扰频器与解扰器符合 CCITT 有关建议
9. 载波检测:
  - 检测电平: 载波关断 -48dBm 或更低
  - 载波接通 -43dBm 或更高
  - 响应时间: 从关断到接通在接收数据到达之前
  - 从接通到关断 发送数据由 Z 极性关断后 5~15ms

10. 在传输时, 保持接收。当 RS 处于接通时, 接收被禁止。如果 RD 由 Z 极性保持或 CD 关断时, 接收也被禁止。

## 二 V. 21 调制解调器

用于传输控制规程的传输, 按照 CCITT T. 4 建议, 三类传真机采用 V. 21 调制解调器的第二信道。

1. 传输速率: 300bps
2. 通信方式: 采用半双工或单工通信方式
3. 调制方式: 频率调制方式
4. 调制速率: 300 波特或更低
5. 线路传输频率:
  - 数据信号“0”: 1850Hz
  - 数据信号“1”: 1650Hz
6. 线路特性要求:
  - 载波段: 0~7 段
  - 负载电缆非线性: 12dB 以下
  - 负载电缆长度: 15 公里以下
7. 传输电平:
  - 发送电平:  $0 \pm 2$ dbm
  - 接收电平:  $0 \sim -40$ dbm

## 三 在公用交换电话中使用的标准化 300bit/s 双工调制解调器(V. 21 建议)

这类调制解调器是为了在普通交换电话网(GSTN)或租用电话电路上进行全双工的低速调制解调器。采用键控调频(FSK)方式, 所用的频率符合 CCITT 建议 V. 21。

### 1. 标称频率

为了在一对交换电话电路或租用电路上实现双工, 采用频带分割的办法, 将信道分为信道 1 和信道 2。在每条信道中, 较高的特征频率( $F_R$ )相当于二进制的 0; 较低的特征频率( $F_Z$ )相当于二进制的“1”

信道 1: 平均频率为 1080Hz, 频偏 100Hz。其特征频率  $F_R$  为 1180Hz,  $F_Z$  为 980Hz。

信道 2: 平均频率为 1750Hz, 频偏 100Hz。其特征频率  $F_R$  为 1850Hz,  $F_Z$  为 1650Hz。

### 2. 特点

1) 在交换电话电路上(或者在租用电话电路上)建立电话呼叫后, 就可以开始数据传输。

2) 在电话电路上数据传输为全双工工作, 正反两个方向上同时以 300bit/s 或低于 300bit/s 的速率进行

数据传输。

3) 数据可以通过同步或异步的方式发送出去。同步传输时, 调制解调器将不必提供在传输停止时维持同步所需的信号。

4) 当两条信道都用于同时双向数据传输时, 信道 1 被用以把主呼者(即通话者)的数据传输至被呼站, 而信道 2 则用于另一方向的传输。

CCITT 建议在三类传真机中, 传输控制规程采用符合 V. 21 的调制解调器的第二信道以 300bit/s 的速率传输。

## 四 公用电话交换网中使用的标准化 4800/2400bit/s 调制解调器(V. 27ter)

这类调制解调器是 CCITT 建议、在公用交换电话网上以 4800bit/s 速率传输数据的调制解调器。

### (一) 特性:

1 采用八相差动编码调制, 以 4800bit/s 的速率传输数据。

2 传输速率能自动降至 2400bit/s, 使用四相差动编码调制。

3 提供一个调制速率可达 75 波特的反向信道, 此信道供选用。

4 包括一个自适应均衡。

### (二) 线路信号

1 载频:  $1800 \pm 1$ Hz, 不另外提供导频单音。

2. 速率 4800bit/s 时, 数据传输速率应为 4800bit/s  $\pm 0.01\%$ , 即调制速率为 1600 波特  $\pm 0.01\%$ 。在发送端, 要发送的数据流被分成三个连续比特(3 比特)为一组的比特组。每一个比特组相对于前一个 3 比特信号码元的相位进行相位变化编码, 3 比特组编码如表 1 所示。在接收端, 对 3 比特码组进行译码, 并按正确的次序重新组合。当这 3 比特码组经过扰码器后, 进入调制解调器的调制器部分时, 3 比特码组左边那个数字是数据流中首先出现的数字。

表 1 3 比特码组相位分配表

| 三比特码组 | 相位变化 |
|-------|------|
| 001   | 0°   |
| 000   | 45°  |
| 010   | 90°  |
| 011   | 135° |
| 111   | 180° |
| 110   | 225° |
| 100   | 270° |
| 101   | 315° |

3. 速率为 2400bit/s 时, 数据传输速率为 2400bit/s  $\pm 0.01\%$ , 调制速率为 1200 波特  $\pm 0.01\%$ 。在发送端, 要发送的数据流被分成两个比特(双比特)为一组的比特组。每个双比特组相对于前一个信号码元的相位进行相位变化编码, 编码与相位的对应关系如表 2。在接收端, 对这些双比特进行译码, 并按正确的次序重

新组合。当这个双比特经过扰码器之后,进入调制解调器的调制部分时,该双比特左边的数字是数据流中首先出现的数字。

表 2 双比特码组相位分配表

| 双比特码组 | 相位变化 |
|-------|------|
| 00    | 0°   |
| 01    | 90°  |
| 11    | 180° |
| 10    | 270° |

**五、点对点四线租用电话型电路上使用的标准化 9600bit/s 调制解调器(V. 29 建议)**

这类调制解调器主要在优质租用电路上使用,当然,也可用于质量较差的电路。

(一) 特性

1 在电路上以 9600bit/s 速率传输数据,且能自动降至 7200bit/s 和 4800bit/s。

2 能够用连续或受控的载频以双工或半双工的方式进行工作。

3 采用幅度和相位相结合的调制同步工作方式。

4 包括一个自适应均衡器。

(二) 线路信号

1 载频:1700±1Hz。不另外提供导频单音。

2 数据传输速率和调制速率

数据传输速率为 9600、7200 和 4800bit/s ± 0.01%;调制速率为 2400 波特±0.01%。

3 信号空间编码

在 9600bit/s 速率时,要发送的数据流,经扰码后,被分成四个连续数据比特(4 比特)为一组的比特组。每个 4 比特组的头一个比特(Q1)用来确定要发送的信号码元的幅度,第二个(Q2)、第三个(Q3)和第四个(Q4)比特相对于前一个码元的相位进行相位变化编码(见表 3)。相位编码与建议 V. 27 规定相同。

被发送信号码元的相对幅度由 4 比特组的头一个比特(Q1)用该信号码元的绝对相位来确定(见表 4),绝对相位由同步信号(训练信号)开始建立。

在接收端,对 4 比特组进行译码,然后数据比特按正确次序重新组合。

当使用 7200bit/s 的减速速率时,要发送的数据流,经扰码后,被变成三个连续数据比特为一组的比特组。比特组的第一个数据比特对应于调制器 4 比特组的 Q2。第二和第三个数据比特分别对应于调制器 4 比特组的 Q3 和 Q4。调制器 4 比特组的 Q1 对每个信号码元均是数据零,其编码与相位的关系与调制器在 9600bit/s 工作状况相同。

当以 4800bit/s 减速速率工作时(表 4)要发送数

据,经扰码后被分成两个连续数据比特为一组的比特组。这比特组的第一个数据比特决定调制器组的 Q2,而第二个数据比特决定调制器 4 比特组的 Q3。对每个信号码元,调制器 4 比特组的 Q1 均是数据零。Q4 由 Q2+Q3 的模 2 和的倒相来确定。信号码元则根据上面所列的表 5 规定来确定。图 4 为以 4800bit/s 的速率工作时发送信号码元的绝对相位图。

表 3 四比特组后的三位码元相位编码表

| Q2 | Q3 | Q4 | 相位变化 |
|----|----|----|------|
| 0  | 0  | 1  | 0°   |
| 0  | 0  | 0  | 45°  |
| 0  | 1  | 0  | 90°  |
| 0  | 1  | 1  | 135° |
| 1  | 1  | 1  | 180° |
| 1  | 1  | 0  | 225° |
| 1  | 0  | 0  | 270° |
| 1  | 0  | 1  | 360° |

表 4 四比特组首位幅度对应表

| 绝对相位      | Q1 | 相对信号码元幅度 |
|-----------|----|----------|
| 0°,90°    | 0  | 3        |
| 180°,270° | 1  | 5        |
| 45° 135°  | 0  | √2       |
| 225° 315° | 1  | 3√2      |

表 5 2 比特组与 4 比特组对应关系

| 数据比特 | 四比特组 |    |    |    | 相位变化 |
|------|------|----|----|----|------|
|      | Q1   | Q2 | Q3 | Q4 |      |
| 00   | 0    | 0  | 0  | 1  | 0°   |
| 01   | 0    | 0  | 1  | 0  | 90°  |
| 11   | 0    | 1  | 1  | 1  | 180° |
| 10   | 0    | 1  | 0  | 0  | 270° |

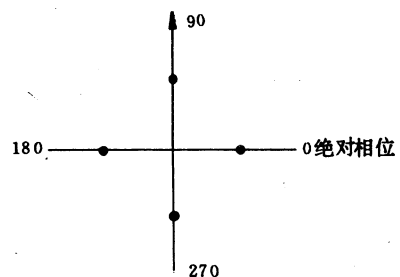


图 4

## 中软总公司库压品门市部

### 优惠向您提供下列产品

#### 一、计算机盘片：

3M. BASF. 五英寸, 单面/双密, 20 元/盒。100 片以上, 15 元/盒; 3M. 五英寸, 单面/双密, 硬分段, 16 扇区, 15 元/盒。CE. BASF 五英寸, 双面/双密, 28 元/盒。100 片以上 24 元/盒。DATALIFE. 五英寸, 单面/双密, 20 元/盒。100 片以上 18 元(塑料盒); DATALIFE. 五英寸, 双面/双密, 30 元/盒。100 片以上 28 元/盒(塑料盒); BASF. MASTER. 八英寸, 双面/单密, 60 元/盒; 3M 八英寸, 单面/双密。80 元/盒; JANUS. 五英寸软加密盘。双面/双密, 45 元/盒(低版本)。

#### 二、盒式磁带：

BASF. 1/2×1200 英尺。60 元/盘;

3M. 1/2×1200 英尺。80 元/盘;

三、8086 单板机, 500 元/台, 9250 智能中英文打字机 4200 元/台; 铅字打字机, 油印机, 日本 1550 九针打印机, 1200 元/台。

#### 四、绘图仪：

SPL—400, 3000 元/台; LP—3700, 2 万元/台; DXY—800, 3000 元/台; 数字化仪, WT—4000, 3000 元/台。

#### 五、显示器：

12 英寸彩显, 800 元/台。12 英寸单显, 400 元/台。

#### 六、各种配置计算机：

IBM PC/XT. AT/286. 中华 386/16. 长城 GW0520CH. GW286.

七、主板机、各种显示卡、长城打印机汉卡; UPS 400W, 1000W

#### 八、其它：

0520A 电源, TP-80 电源, 打印机电缆线, 光电鼠标器——100 元/个, 稳压电源 100W 扁线压接工具、实验版、起拔器、1500K 盒式色带、P3/P7 盒式色带、元器件; 5 英寸、8 英寸盘盒。各种色带, 13×10 正反环, 13×15 开环等。

联系人: 温友良 孟秀华

电话: 8317722—1112

地址: 北京市海淀区学院南路 55 号

电脑大厦 112 房间

邮政编码: 100081

乘车路线: 16 路汽车电脑大厦下车

## 全国电子报刊联合征订简明目录表

| 订阅代号   | 报刊名称         | 刊期  | 开本页码         | 单价   | 全年订价      | 通讯地址                   | 邮码     |
|--------|--------------|-----|--------------|------|-----------|------------------------|--------|
| 1—28   | 计算机世界报       | 周报  | 4 开 160 版    | 0.48 | 24.00     | 北京市 750 信箱             | 100039 |
| 1—48   | 北京电子报        | 周报  | 4 开 4 版      | 0.12 | 6.24      | 北京朝阳区东三环北路 36 号        | 100026 |
| 1—49   | 国际电子报        | 周报  | 4 开 32 版     | 0.25 | 12.00     | 北京市 750 信箱             | 100039 |
| 1—145  | 电子商报         | 周二  | 对开 4 版       | 0.20 | 20.80     | 北京石景山路 23 号            | 100043 |
| 2—145  | 家用电器         | 月刊  | 16 开 32 页    | 0.72 | 8.64      | 北京西城区月坛北小街 6 号         | 100037 |
| 2—354  | 电视技术         | 月刊  | 16 开 64 页    | 1.70 | 20.40     | 北京 743 信箱              | 100015 |
| 2—675  | 电信技术         | 月刊  | 16 开 48 页    | 1.30 | 15.60     | 北京东长安街 27 号            | 100740 |
| 2—888  | 电子与电脑        | 月刊  | 16 开 56 页    | 1.60 | 19.20     | 北京 173 信箱              | 100036 |
| 2—889  | 电子技术应用       | 月刊  | 16 开 48 页    | 1.50 | 18.00     | 北京 927 信箱编辑部           | 100083 |
| 2—892  | 电子世界         | 月刊  | 16 开 32 页    | 0.75 | 9.00      | 北京市 165 信箱(万寿路西街 11 号) | 100036 |
| 4—141  | 电子技术         | 月刊  | 16 开 48 页    | 1.50 | 18.00     | 上海 085—253 信箱          | 200009 |
| 4—236  | 中学科技         | 月刊  | 16 开 48 页    | 1.20 | 14.40     | 上海冠生园路 393 号           | 200233 |
| 4—316  | 现代通信         | 月刊  | 16 开 32 页    | 1.00 | 12.00     | 上海 085—253 信箱          | 200009 |
| 4—386  | 无线电与电视       | 双月刊 | 16 开 48 页    | 2.00 | 12.00     | 上海市瑞金二路 450 号          | 200020 |
| 4—389  | 实用无线电        | 双月刊 | 16 开 48 页    | 1.30 | 7.80      | 上海市冠生园路 393 号          | 200233 |
| 46—115 | 电脑           | 月刊  | 16 开 80 页    | 1.50 | 18.00     | 广州石牌华南师范大学内            | 510631 |
| 61—8   | 西部电子信息报      | 周报  | 4 开 4 版      | 0.15 | 7.80      | 成都市桂王桥西街 66 号          | 610017 |
| 61—74  | 软件报          | 周报  | 4 开 4 版      | 0.18 | 9.36      | 成都市金河街 75 号            | 610015 |
| 66—32  | 机械与电子        | 双月刊 | 16 开 48 页    | 2.00 | 12.00     | 贵州贵阳市延安西路 67 号         | 550003 |
| 77—19  | 电脑报          | 周报  | 4 开 4 版      | 0.15 | 8.16      | 重庆市双钢路三号               | 630013 |
| 82—141 | 国外电子测量技术     | 季刊  | 16 开 48 页    | 2.00 | 8.00      | 北京 2452 信箱             | 100083 |
| 82—339 | 计算机世界月刊      | 月刊  | 16 开 120 页   | 2.00 | 24.00     | 北京 750 信箱计算机世界月刊部      | 100039 |
| 82—417 | 微型机与应用       | 月刊  | 16 开 48 页    | 1.50 | 18.00     | 北京 927 信箱编辑部           | 100083 |
| 自办发行   | 电脑爱好者        | 月刊  | 16 开 56 页    | 1.80 | 21.60     | 北京海淀区中关村南二街五号 102#     | 100080 |
| 自办发行   | 电子天府         | 双月刊 | 16 开 128 页   | 4.00 | 24.00     | 成都市育婴堂街 20 号           | 610017 |
| 自办发行   | 今日电子         | 月刊  | 大 16 开 96 页  | 6.80 | 40.00 含邮费 | 海淀区车道沟一号滨河大厦九层         | 100081 |
| 自办发行   | 电讯技术         | 双月  | 16 开 90 页    | 4.00 | 24.00     | 四川省成都市 94 信箱           | 610036 |
| 自办发行   | 电子质量         | 月刊  | 大 16 开 96 页  | 2.68 | 33.60     | 广州 1501 信箱 9 分箱        | 510610 |
| 自办发行   | 广东电子         | 月刊  | 大 16 开 48 页  | 2.50 | 30.00     | 广州 1501 信箱 9 分箱        | 510610 |
| 自办发行   | 电子产品可靠性与环境试验 | 双月刊 | 16 开 72 页    | 3.00 | 18.00     | 广州 1501 信箱 9 分箱        | 510610 |
| 61—75  | 电子报          | 周报  | 4 开 8 版      | 0.22 | 11.52     | 成都市金河街 75 号            | 610015 |
| 自办发行   | 音响世界         | 月刊  | 大 16 开 116 页 | 6.50 | 78.00     | 广州市石牌五山路科技街 108 号      | 510630 |
| 62—175 | 实用电子文摘       | 双月刊 | 16 开 126 页   | 2.80 | 16.80     | 成都市金河街 75 号            | 610015 |

# 双频卡使用问答

程 传

PC 机的各种选件卡中,双频卡的使用是较灵活的。为帮助缺乏技术资料的用户更好地使用双频卡,我把实际安装、使用中常见的问题整理成以下问答,供广大读者参考。

## (一)PC 机有哪些显示器适配卡?

PC 机一般使用单色显示器、彩色显示器或家用电视机作为显示器。与此相适应的适配卡分别是单色显示器适配卡(单显卡)、彩色显示器适配卡(彩显卡,又分 CGA 卡,EGA 卡,VGA 卡等)和复合视频信号卡(视频卡,或称 TV 卡)。

早期生产的显示卡(俗称长卡)一般只能适用一种显示方式,有的设有一个同轴的 TV 插口,可直接连接电视机。现在常用的显示卡功能都较多,如双频卡,能用于几种显示方式,并附带了其他功能。

本文介绍的主要是双频卡的功能和使用方法。

## (二)主板应如何和显示卡适配?

各种显示方式都必须配用相应的显示卡,而双频卡能一卡多用,所以很受用户欢迎。但是,不论选用何种显示卡,主板的设置也都必须和显示方式一致。目前,PC 机常用的主板有主频 4.7MHz 的旧式主板(俗称大板)和最高主频 8—12MHz 的新型主板(俗称小板)二种。不管是哪种板,板上都有一个系统状态设置开关,是 8 位 DIP 开关。其中,选择显示器接口类型的开关是 SW5、SW6。开关设置表如下:

| SW5 | SW6 | 显示器接口方式    |
|-----|-----|------------|
| ON  | ON  | 不接显示器      |
| OFF | ON  | CGA(40×25) |
| ON  | OFF | CGA(80×25) |
| OFF | OFF | 单色显示方式     |

大板上的 DIP 开关较多,调整时可参看有关说明或由所在位置判断出系统状态设置开关。

## (三)双频卡有哪些功能?

常见的双频卡可适用三种显示方式:

1. 彩色图形显示方式(CGP)
2. 单色显示方式(MDP)
3. Hercules 显示器方式

此外,还可在单色显示器上模拟彩色显示;可以输出复合视频信号,用电视机作显示器,还有自动开关和光栅自动熄灭功能等。

## (四)如何使用双频卡上的插口?

双频卡上最主要的是显示器插座(DB9 型),位于安装板上部。此外,大部分双频卡还有打印机插座,

(DB25 型),一般位于安装板下部。

此外,卡上一般还有复合视频信号插座,使用时需另接同轴插座输出。有的卡还附有光笔接口,游戏接口等。这些接口的 IC 有时只备有插座,用户要使用时,再自行插上,例如,使用游戏接口一般要插上 74 LS244 和 558 二块 IC。

## (五)怎样选择双频卡的工作方式?

双频卡的工作方式由一个 8 位的跳线开关选择决定。8 位开关的边上还有一个单位开关,用来在 CGA 方式时选择字型。

JP1、JP2 用作打印接口选择,接法如下表:

| JP1 | JP2 | 打印接口    |
|-----|-----|---------|
| 通   | 通   | 不使用     |
| 通   | 断   | 选用 LPT1 |
| 断   | 通   | 选用 LPT2 |
| 断   | 断   | 选用 LPT3 |

其余各位开关作用如下表:

| 编 号 | 项 目                 | 接 通        | 断 开        |
|-----|---------------------|------------|------------|
| JP3 | Hercules/CGA 显示方式选择 | CGA        | Hercules   |
| JP4 | 自动开关方式设置            | 不选用        | 选用         |
| JP5 | 字符发生器 ROM 选择        | 单字符<br>ROM | 双字符<br>ROM |
| JP6 | 自动熄灭光栅方式设置          | 不选用        | 选用         |
| JP7 | 模拟彩色显示方式设置          | 不选用        | 选用         |
| JP8 | 保留脚                 |            |            |
| JP9 | CGA 字型选择            | 7×7 型      | 5×7 型      |

必须注意的是:

1. 选用模拟彩色显示方式时,不能使用 CGA 显示器。

2. JP8 只能处于“接通”状态。

## (六)如何把单显卡设置成单显卡或 CGA 卡?

双频卡在出厂时一般设置在自动开关方式,用户需设置成单显卡或 CGA 卡时,可按以下方式跨接:

1. JP4 接通(设置于非自动开关方式)。
2. JP3 断开时,置单显方式;接通时,置 CGA 方式。

(七)用彩色电视机作显示器时,为什么不能显示彩色?

CGA 卡受原设计限制,即使使用彩色电视机作显示器也只能显示单色字符。但有些双频卡在这方面已作了改进,能显示彩色字符。

## (八)什么是彩色模拟显示?

彩色模拟显示是在单色显示器上,用不同灰度来模拟不同的颜色,以获得类似彩色的效果。使用这种显示方式时,必须改变主板上的系统状态设置开关或通

过软件切换,把系统设置在 CGA 方式。

(九)什么是软件开关?它有什么优点?

PC 机系统上电之后,系统的 BIOS 将检查主板上系统状态开关的设置状况,并把它和所用的显示卡进行比较。如发现异常情况,显示器将显示诸如“display card doesn't match(显示卡不匹配)”等警告信息。在自动开关方式下,双频卡能“感知”主板上的设置情况,自动地调整到相同的显示方式状态,从而不会再出现出错信息。有了这种功能,最终用户就不必再为开关状态的设置而头痛了。

(十)什么是软件开关?

软件开关指用户能够在开机后,使用由 DOS 提供的 MODE 命令来改变显示卡工作模式的方法。例如:用“MODE CO80”可选择 80 列的彩色显示方式,用“MODE CO40”可选择 40 列的彩色显示方式,用“MODE MONO”可选择单色显示方式。

(十一)如何使用双频卡上的打印接口?

PC 机开机后,BIOS 将按 3BCH、37CH、27CH 的顺序查验整机系统里有几个打印接口,并按此顺序给每一个打印口分配一个编号(LPT1、LPT2、LPT3)。例如:

1. 主板上已有一个地址为 37CH 的打印口,双频卡上跨接线设置打印口为 27CH,则 BIOS 将定义:

主板上的打印口(37CH)为 LPT1,  
双频卡上的打印口(27CH)为 LPT2。

2. 主板上已插有 2S—2P 卡,其打印口地址设置为 3BCH、27CH,则双频卡上的跨接线应设置打印口为 37CH,这时 BIOS 将定义:

2S—2P 卡的打印口为 LPT1 和 LPT3,

双频卡的打印口为 LPT2。

3. 若整机只双频卡上设有打印口,则无论其地址是哪一个是, BIOS 都把它定义为 LPT1。

(十二)请举出几个设置实例。

下面以列表形式给出设置实例,供使用者参考。表中未列出的显示卡的跨接线有 JP1、JP2、JP5(在各实例中都“接通”),JP6(用法见第三问),JP8(保留位)。这 5 位的接法和显示方式关系不大,故未列入表中。

| 显示方式       | 主板设置 |     | 显示卡设置 |     |     |     |
|------------|------|-----|-------|-----|-----|-----|
|            | SW5  | SW6 | JP3   | JP4 | JP7 | JP9 |
| 单色显示       | 断    | 断   | 断     | 通   | 任意  | 任意  |
| CGA(40×25) | 断    | 通   | 通     | 通   | 通   | 注 1 |
| CGA(80×25) | 通    | 断   | 通     | 通   | 通   | 注 1 |
| 模拟彩色显示     | 注 2  | 注 2 | 通     | 通   | 断   | 任意  |
| 复合视频信号输出   | 断    | 通   | 通     | 通   | 通   | 注 1 |
| 自动开关方式     | 注 3  | 注 3 | 任意    | 断   | 注 4 | 注 1 |

注 1:通,7×7 字型;断,5×7 字型。

注 2:可以 SW5 通、SW6 断或 SW5 断、SW6 通。

注 3:SW5 和 SW6 的状态可以任意组合,但不能同时“断”。

注 4:通,模拟彩色显示方式;断;非模拟彩色显示方式。

以上介绍的是双频卡在 PC 机上的用法,实际上它也可用于 286 机,除主板设置由软件进行外,其他步骤都是相同的。

## 两处疏漏一点建议

何崇乐

《电子与电脑》#12/92 载文《读者联谊(十二)》(下简称《王》文)的 52、53 两条,存有不妥之处;特指出,请读者尤其是“机友”留意。

1 关于从 1.2M 盘向 360K 盘拷贝的问题,

(1)若 1.2M 盘上的内容少于 360K 字节,则完全可以视 1.2M 盘为 360K 源盘,来进行所需的拷贝。

(2)若 1.2M 盘上的内容多于 360K 字节,则仅在 DOS3.X 下,按《王》文的方法是可行的;但在 2.X 下就行不通了。换言之,这种方法的实施是有条件的,因 RESTORE 命令在 2.X 下不支持硬盘向软盘的恢复。而现在,在 DOS 2.X 下工作的人,还是有的吧,这种情况下,最好借助于 PC Tools 等工具软件来分批拷贝。

2 关于“Missing Operating System”死机。

《王》文的方法见《电子与电脑》#12/92(P.42 右栏上部),

(下转 49 页)

北京电子工业出版社《电子与电脑》杂志编辑部将于 8 月 30 日开始举办为期四周的微机硬件二级(芯片级)维修技术学习班

培训内容:8088、80286 主机板和 80386、80486 简介;3.5 英寸、5.25 英寸软驱(高低密度)及软卡;硬盘硬卡;CGA、EGA、VGA 显示器;小型不间断电源(UPS)等工作原理及实用维修技术,并通过演示课练习排除故障,并招一学期代培生两名。

上课地点、食宿(费用自理)均在人民大学校园内。

培训费:410 元

资料费:实收(约 120 元)

联系人:北京中国人民大学信息中心 胡野红 梁杰熙

邮编:100872 电话 2555431 转 3305