

E&C

1994

●一九九四年 ●总期第107期

2

電子

ISSN 1000-1077

與 電腦

电子工业出版社广州科技公司大量提供

单片机应用系统及各类电子控制 设备专用的开关电源



• ELECTRONICS AND COMPUTERS •

● 超前市场牵引 ● 电子技术推动

威达实业公司



值此庆祝威达实业公司(原潮阳县海门威达电子器材经营部)开业 10 周年暨被《家用电器》杂志连续三年评为邮购服务先进单位之际,综合本公司 1986 年至 1993 年度在《无线电》、《家用电器》、《家电维修》、《电子世界》等刊出广告中的全部新老产品,本着薄利多销原则,开展邮购服务。邮购下列产品(含本期封三所列产品),请参考上述杂志有关广告或函索价目表(价目表工本费邮资 0.20 元)。邮购汇款附言内须注明产品名称和数量。注明详址和邮编。

图(1)A, 100W+100W 全自动倒立放扩机:日本进口自动电脑选曲机芯 自动翻转连续放音 二路 BBD 卡拉 OK 伴唱 DC 对称放大电路 多级恒流 前后级全段直接耦合设计 每台: 365.00 元 邮资: 30.00 元

B, DK-988 立体声降噪双卡式放音卡座:杜比噪音抑制系统 卡式座 1 及 2 自动倒带连续播放 METAL/NORAL 1 及 2 卡式座磁带选择键功能 常速、快速同步录音复制 多种信号输入及卡拉 OK 电脑选曲功能 每台: 560.00 元 邮资: 30.00 元

C, 卡拉 OK 立体声收音卡式录放机: FM/MW 收音、录音、放音、卡拉 OK (BBD)、扩音,自动频率控制 音频三段均衡 功率: 2000mW(RMS) 每台: 245.00 元 邮资: 15.00 元

图(2)A, DK986 立体声降噪双卡式录放音座(性能同图(1)B 体机,无自动倒带功能) 每台: 420.00 元 邮资: 30.00 元

B, 120W+120W 多功能卡拉 OK 伴扩机:三路 BBD 环绕声卡拉 OK 影碟、唱机、录像、录音信号输入/输出 每台: 290.00 元 邮资: 20.00 元

C, 双十段仿富丽高级液晶频谱均衡器:大液晶显示屏 数字集成电路 多级恒流电路 DC 频响效果 每台: 305.00 元 邮资: 20.00 元

D, ST 型立体声调谐器(收音机): FM, AM, SW₁, SW₂ 四波段 调谐指示和静噪功能 场效应管高频放大 FM, 低阻式磁场天线 AM TA7640 集成电路中放 售价: ST 型 4 波段每台 255.00 元 ST 型 2 波段每台 230.00 元 邮资: 20.00 元

* 图(2)A 至 D 组合配二只 12 吋以上音箱即构成一套高档组合音响, ABCD 1200.00 元/套 邮资: 80.00 元。

广东省潮阳市威达实业公司

地 址: 广东潮阳市海门镇北门城外大街 43-2 号

邮 编: 515132

电 话: (07644)631390

电 挂: 4241

联系人: 郑良加

开户行: 潮阳市工商行海门办

帐 户: 潮阳市威达实业公司

帐 号: 066024

常年法律顾问: 广东省潮阳市海门镇法律服务所 陈廷顺。

● 国内代号: 2-888 定价: 2.40 元

单片机应用系统及各类电子控制设置专用

开关电源

规 格	零售价	邮挂费
5V/10A+12V/1A, -12V/1A; 外表尺寸: L158×W98×H48mm ³	180 元/台	7 元/台
5V/15A(LED 大屏幕显示用) 尺寸同上	180 元/台	7 元/台
5V/10A, +12V/1A, -5V/2A 尺寸同上	180 元/台	7 元/台

1. 订购 10 只以上为批发, 价格便宜 5%。
2. 可根据使用要求, 生产专用规格的电源。

特性:

SUNSHINE WY09-3B 开关电源

输入电压: AC220V \pm 15% 0.7A 50Hz

AC110V \pm 15% 1.4a 60Hz 开关切换

输出电压:

+5V 端: 恒压输出, 其可调范围 4.5V~6.5V, 主负载输出端

工作电流 最大值 15A

纹波电压 小于 100mV

瞬变特性 100%负载变化时优于 \pm 0.5%

+12V 端: 恒压输出电压误差范围 \pm 2%

工作电流, 最大值 1.5A

纹波电压 小于 50mV

瞬变特性 100%负载变化时优于 \pm 2%

-12V 端: 恒压输出 电压误差范围 \pm 2%

工作电流 0~0.5A

纹波电压 小于 50mV

瞬变特性 100%负载变化时优于 \pm 2%

输出功率: 75W 最大值 90W(需加 2mm 水柱高风压散热)

空载功率: 小于 6W

工作效率: 大于 82%

开机涌浪: 小于 20A/10ms

隔离电压: AC、DC、FG 间大于 1000VAC1Min
TEM.=27° HUM=60%

绝缘电阻: AC、DC、FG 间大于 100M Ω TEM=27°
HUM=60%

工作环境: TEM=-10℃~+35℃ HUM=10%~90%REL

工作温升: 在+27℃自然无风冷散热情况下, 机身温度低于+70℃

外形尺寸: L158×W98×H48mm

机身重量: 小于 0.5kg

本机还具有交流滤波, 开机缓冲冲击、过载保护、输出输入短路保护、LED 工作指示、外壳与地隔离等功能。

主要应用于微处理机, 可编程控制器、高精程控制器、高精密电子仪器、开发机, 显示屏供电等。

公 司 地 址: 广州市天河五山路华南大学科技大楼 215 室

邮编: 510631

开户行及帐号: 广州建行天河高新办 480-261062-09

电子工业出版社广州科技公司 联系人: 陈昊

电话: (020)7588476, 7588494, 7588493, 7504488-3872

名称及说明	单价	邮资	“珠江、国光、银笛”系列扬声器	单价	邮资	93 音响 Hi-Fi 发烧级套板件(邮资均 10 元/次)
MF-110A 型袖珍超小型万用表: 具直流电流 0-500mA, 电压 0-1000V, 交流电压 0-1000V, 电阻 $R_{10} \times 100 \times 1K$, 音频 -20+62dB 等 5 级精度量限。	40	5	2 寸 8Q4W 高音 2 寸 8Q10W 高音 3 寸 8Q5-10W 高音 4 寸 8Q4-8W 中音 4 寸 8Q3W 低音 5 寸 8Q5W 低音 5 寸 8Q5-10W 中音 5 寸 8Q 球顶中音 20W 6 寸 8Q8W 低音(白盆泡沫边) 8 寸 8Q10-30W 低音(白盆泡沫边) 10 寸 8Q30-50W 低音(白盆泡沫边)	1.5 3.2 4.5 7 4 10 15 27 15 32 48	2 2 3 3 3 4 5 5 10 10 15	●双十段均衡板: 超静噪 B 管 20 支作模拟电感, 高速场效应运放 082×2 作均衡中央控制器, 前级设立立体声/全景环绕声电路, 整流滤波双组稳压集成电源, 单联直滑式电位器, 全板一体化, 音质纯真通透, 成品 60 元, 配套用面板按钮 10 元。●全景环绕声器: TL082 倒相声场变幻变化, 成品 15 元。●卡拉 OK 混响放大板: (40W+40W), 功放采用 HA 或 TDA 集成工作, 中心混响以 MN3102、3207, 不用调试接上电源即可工作, 成品板 60 元, 半成品板少 MN3102、3207 集成, 备 IC 插座 35 元。●双十段均衡音调板: 用 TA7796 或 LA3600 四只直滑 20 只装 40 元, 配面板手钮加收 5 元。●带散热整流前置高低音调节成品扩音板 TD2030 四只装 2×50W45 元, 成品整机 80 元/台; TDA2030 二只装 2×30W 板 28 元, 成品整机 55 元/台单 30W 板(大功率管装)8 元, 成品整机 28 元/台●268(改进型, 卡拉 OK 回响混音前置放大板, 磁头放大以 LA3161, 中心混响以 MN3102、3207 五电位器调节, 可直接推动扩音机 68 元。)
YOC 液晶数字显示测电笔: 可直接测检 6-220V 交直流电和间接测检交流电的零线断点, 数字直读显示。	25	5	10 寸 8Q100W(泡沫边) 12 寸 8Q40-60W(泡沫边) 音箱分频器: 珠江 2 分频 20-40W 音箱分频器: 珠江 3 分频 40-100W 音箱分频器: 珠江 4 分频 40-60W 1.6 米宽幅黑色尼龙音箱布(每米) 8 寸 8Q 玻璃纤维低音 100W 8 寸 8Q 碳纤维低音 100W 10 寸 8Q 玻璃纤维低音 100W 10 寸 8Q 碳纤维低音 100W	80 95 12 20 40 10 195 240 240 285	15 15 4 4 4 3 10 10 15 15	
超小型立体声收音机: 体积 7.5×5×1cm 带耳机、单 FM 波段双 AM/FM 波段	28 40	5 5				
高级自动对讲电话机: 联络距离高达 2000 米以上, 使用时只要一按开关, 即发出与电话铃相同讯号, 接受一提话筒即可通话, 放下自动关机停电, 专利产品。	60	5				
全自动卧式放扩机: 具进口机芯, BBD 回响双话筒混音, 双五段均衡集成电路, 按钮星光指示, 能自动翻转连续放音, 功效用大功率管 OCL 电路装, 功率: 80W+80W	28	20				
100 米无线调频话筒: 频率 FM88-108MH	15	5				
FD 有线动圈话筒:	17	5				

遥控 IC 专柜 (PTC 音响系列 IC):

型号	特性	说明	单价	邮购办法说明: 我公司专柜供应遥控 IC 系当今最新集成器件, 除本期刊出的“PTC 音响系列 IC”外, 还备有系列: PTC 编解码, PTC 风扇控制器等。为方便厂商与个人电子爱好者的生产设计及使用, 本公司备有详细全部资料和部分应用电路图, 用户可按图组装也可参考自行设计。欢迎购买或来信咨询, 生产厂家批量购买选用按表中零售邮购单价优惠 15%, 量大请来电商谈。
PT2248A/ PT2249	宽范围工作电压, $V_{CC} = 1.5 - 5.0V$ CMOS 工艺, 低功耗复合按键可以(多达 6 个)最少的外围元件由于振荡电路内藏, 外接 LC 或陶瓷晶体即可构成振荡器	PT2248 是采用 CMOS 技术为红外遥控发射器用而开发的大规模集成电路。它有 18 种功能和 75 条指令。有单发射键, 连续键和复合按键。PT2248A 与 TC9148 全兼容。	20.40 元	
PT2249A/ PT2250A	可以从发射器送出多路同时按键信号(PT2249A 可即时送出 5 种功能而 PT2250A 可同时送出 6 种功能)可提供单脉冲, 保持脉冲和循环脉冲输出。(循环脉冲只有 PT2250A 能做到)用 CR 构成振荡器采用码检测电路防止来自各种仪器设备的干扰很强的噪声抑制能力	PT2249A/PT2250A 是为红外遥控接收器用而研制的大规模集成电路。当它与 PT2248 发射器结合使用时, 可构成遥控系统。PT2249A 是 16 脚 DIP 封装, 可控制 10 种功能而 PT2250A 是 24 脚 DIP 封装可控制 18 种功能。PT2249A、PT2250A 与 TC9149、TC9150 全兼容。	23.60 元	
PT2395	PT2395 与 2248、2249、2250 不同的二大特性: PT2395 则已将 74HC04 放入 IC 内部, 替设计生产厂商省下外接 74HC04 的成本与麻烦。另一项特点是允许厂商外接 61256. DRAM (256KBIT) 而得到 4 倍, 即 786.4mS 的延迟时间, 而产生更好的回音效果。当然 6164DRAM 也可直接使用。	PT2395 与 PT2248A、PT2249A/2250 等全兼容。	40.05 元	

电子与电脑

一九九四年
总期第 107 期

目 录

• 综述 •

- 混合输入法介绍 陈德发(2)
三级学习机的设想 张 挺(2)

• PC 用户 •

- 浅析 Turbo C 图像处理函数的缺陷与像素的保存和恢复
..... 屈景辉、任京芳(4)
一个多功能的硬盘主引导程序 燕士海(6)
四通 MS-2401 与北大方正排版文件的转换
..... 王景余(8)
谈谈自由组合条件查询语句的形成 庄 志(9)
Keyboard 命令在全屏编辑中的使用 沙 玮(10)
一种汉字无级放大的方法 陈德运、李秀忠(12)
倍受青睐的 DR DOS 6.0 李瑛彬(13)
扩展或扩充内存的使用——多任务操作 ... 仲云辉(16)
2709 病毒的分析和诊治 万武庆(17)

• 学习机之友 •

- 中华学习机做简易字幕机 李建林(18)
ProDOS 系统内部结构剖析 廖凯(21)

• DOS 操作系统分析讲座 •

- 第二讲 键盘输入系统(上) 崔来堂(23)

• 学用单片机 •

- 一种扩充单片机程序存储器的巧妙方法 ... 王仕仲(26)

• 电脑巧开发 •

- 普及型逻辑笔的制作 车金相(25)

- 8086 接口实验系统设计 江 琪(30)

• 维修经验谈 •

- PC 机多功能控制卡特殊故障维修一例 刘绍华(37)

• 电脑游戏机 •

- 第五章 背景画面的位移操作(上) 于 春(38)

• Windows 技术讲座

- Microsoft Windows 3.1 应用基础——基本操作(上)
..... 徐鹏力(42)

• 初学者园地 •

工具软件 PC Tools 的应用

- 二 PC Tools 文件服务功能 王路敬(45)
谈谈 COMMAND.COM 王伟廷(47)

• 电脑通信 •

- 扫描仪输入技术及应用 ... 方卫宁、何晓波、邹 华(48)
利用 PC Tools 7.0 实现 PC 机间的高速通信
..... 程启超(50)

• IC 电路应用 •

- 交流有效值一直流变换 IC AD736/AD737 及应用
..... 王俊省、李兰友(52)
Intel 486 系列 CPU 综述 吴 胜(54)

• 读者联谊 •

- 使用 DOS 6.0 经验一则 陈乃强、李 志(51)
• 回音壁 •
也谈 WPS 文件阅读器 张国栋(56)

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:2.40 元

混合输入法介绍

陈德发 北京兄弟电脑公司

由于键盘上的键数有限,要想输入成千上万个汉字,每个键只代表一个汉字是远远不够的,因此必须将汉字按照一定的规则用两个或两个以上的字母来代表,这个规则就是所谓的编码法。目前汉字的编码方法虽有成千上万种,但一般说来可分为专业和非专业汉字录入人员使用的两大类,汉字的输入速度与其易学性易记性是矛盾的,如:五笔字型汉字输入法,就是需记忆的规则多,掌握后还必须经常使用,否则会忘记,但由于掌握后输入速度较快,因此有很多专业汉字录入人员使用。许多非专业汉字输入员都在使用拼音法输入汉字。汉语拼音作为学习汉字的辅助手段,是我国语言工作的一项基本国策,因此,绝大多数人在小学时就掌握了汉语拼音,且终身不忘。但用标准拼音输入汉字重码太多且每个汉字用的代码也太长,因此需改良创新。早在80年代电子部六所开发的CCDOS汉字系统就将一些冗长的拼音声母及韵母用一个字母代替,以减少码长,代换规则如下:

标准:ch sh zh ai an ao en ang eng ing ong ü

简化:i u a l j k f h g y s v

这套代码规则在全国乃至全世界流传甚广,影响最大。用这种简化的拼音输入汉字人们仍然觉得太慢,可又毫无办法。本人曾有幸在晓园语文与文化研究所工作过,发现汉字中以双音节的二字汉字词组使用频率最高,这一点随便看一篇文章便可验证;又发现将一个词组看成一个汉字再用拼音编码后重码率便会大大地降低,即使有一些重码,由于两个汉字比单个汉字好认或者说识别率高,也很容易选择。根据上述理论与国情,本人发明了混合输入汉字的方法,该方法的精髓是:以拼音为基础,尽量减少汉字输入时的各种转换,充分利用人们已学习过的拼音输入汉字的方法,在不用学习新规则的情况下,快速输入汉字。该输入法已申请国家发明专利。

词组编码规则:

二字词组:编码为该词组简拼拼音码。当字的简拼拼音码多于两个时,只取前两个,如:“中国”词组编码为“vsgu”(拼音声母zh用v代表,其它同上面CCDOS的简化规则)。“伟大”的词组编码为“weda”。

由三个或四个汉字构成的词组简称三字词或四字词,编码规则为该词组中每个汉字的第一个拼音码,如:“解放军”的编码为“jflj”,斗志昂扬为“dzay”。

如何使用混合汉字输入法:

混合输入就是将已知的区位、标准拼音、简化拼音、双拼拼音、词组及联想输入都集中到一个汉字输入

状态(ALT F3)中去,在此状态下:如果输入阿拉伯数字,则按区位码输入汉字;如果输入的是字母,则按标准拼音或简写拼音输入汉字;如果输入的是字母加空格,则按该字母代表的词组输入汉字,重码汉字或词组分别以数字区分。

不论用上述那种方法输入的汉字,都可以联想出与该汉字有关的联想词组。

举例,在Alt F3汉字状态下,如输入“中”字:(1)可输入区位码:“5448”; (2)可输入标准拼音:“zhong”; (3)可输入简化拼音:“vs”; (4)如输入词组“中国”则输入:“vsgu”,而且每种方法都可以同时具有联想输入的功能。

在该系统中,已经用硬件固化了一万多条常用词组,包括汉语成语字典、三字词和二字词,这些词组也不占任何内存,用户还可随时添加自造词组,并保存下来。

计算机应用越来越广泛,必将渗透到人们的各个生活与工作领域中去,因此,会有越来越多的人士使用计算机,让他们都学习专业汉字录入的汉字输入方法显然是不行的,混合输入法正好与您的现有知识衔接,再稍加学习,便可无师自通,达到熟练输入汉字的目的,所以,该输入法有广阔的应用领域和巨大的社会效益。

三级学习机的设想

张挺 广州

中华学习机已有十年历史了,更新换代的问题自然也提到日程上来。笔者认为分开三级学习机替换中华学习机较好。学习机的含义,无非是为用户提供学习的上机环境,普及计算机知识,体现邓小平同志提出“计算机的普及要从娃娃抓起”的要求。学习机的用户是广泛存在的。除了一般的计算机低层次专业用户外,还有广大对计算机事业认可和支持的国人,没有计算机知识又想使用计算机的人们。三级的学习机,较能适应各种各样的用户。

所谓三级计算机,就是软件类似,操作系统不同,硬件完全不同的高,中,低档三种计算机。特点是档次拉开很大,借鉴国外的软件体系,在高级软件中统一规

范,方便软件的移植,主要是在标准程序语言上增加 I/O 格式、控制符、初始化、工作状态等方面的标准。符合标准可直接移植,不符合标准的软件在各种特定的领域中起作用,标准内的软件 and 标准外的软件都有发展的领域,其中三级学习机共同使用标准软件体系发展,高级学习机使用 IBM-PC 软件体系发展,中级学习机使用 Apple IIe 软件体系发展,低级学习机使用任天堂软件体系发展。用户使用计算机完成学习任务外还有其他用途,如文字处理、财务管理、智力游戏、外语翻译、电路模拟、CAD 制图等,非标准软件适合这些用途。按中国的国情,学习机应该中国的个人计算机,不必象国外那样强调计算机的高性能,不必严格区分家庭计算机,学习计算机,台式计算机的差别,我们发展的计算机要求一机多用,物尽其用,非标准软件从中起很大的作用。这里的三级学习机不偏重标准或不标准的软件。与中华学习机相比,设想中的三级学习机强调更高的附加值,从更多的方面满足用户的要求。

学习机的市场是一个大市场,各种各样的用户具备计算机知识不同,分开低级、中级、高级的学习机,会使简单计算机的顾客,文字处理计算机的顾客,程序开发及其他专业工作的顾客各有所爱,各级学习机有通用的学习软件,顾客只需根据最终的用途购买学习机,这样,三级的学习机提供合理的选择空间。

低级学习机针对青少年儿童爱玩特点,增加学习的乐趣。这方面任天堂体系游戏机较好,只是学习用的软件不足。结合中华学习机的软件和游戏机的软件运行是低级学习机的出发点,广东惠州 CEC-2000 就是一个例子,不过笔者认为要改进。中华机开关电源,游戏卡都是高成本的,对一些只要简单计算机的用户来说不合适。PEC-9386 的情况好一些,特别是 36 引脚扩展卡较好,Apple 计算机常用的引脚是 34 左右,36 引脚是一个好规范,由于总线没有保护,各厂商可以采用这一标准,低级学习机从这里展开。

笔者设想的低级学习机是这样的:使用中华机的键盘格局,四个移动键处改为游戏键区,适合文字学习和游戏娱乐的操作;不用软盘,改为 128K 以上的内存,可以较好地运行虚拟操作系统,游戏的内存空间也在其中;使用 36 引脚多个扩展槽,可以用 SRAM 保存必要的数据进行记事操作、扩展系统的功能、提供二次开发的空、固化高价的程序;把磁带作为基本的接口,磁带程序保护的可能性较小,便于交流,接口操作较繁,可限制儿童游戏上机,价格低,且高速化的磁带格式结合超大的内存,利大于弊;系统不用开关电源,成本低,简单实用。其关键是软开关驱动程序,使三套译码体系正常工作,有相应的硬件配合,分核心游戏空间,中华机空间,公共内存外空间。

中级学习机针对中小学的课程而设计,价格不能过高,是实现所谓“学校手动教育,家庭无纸教育”的理

想而设计的。中华学习机美国也有类似的机种,只是 CP/M 扩展缺点多,运行不理想。由于 CP/M 有软件总线之称,内部结构非常规范,与 IBM-PC 的操作系统相近,如何在中级学习机中低成本运行 CP/M 体系软件和中华机体系软件是设计中级学习机的出发点。目前笔者所知台湾的 W.W 兼容机是 6502cpu 和 Z80cpu 结合在一起,只是技术过时了,新的学习机应适当使用新技术,中级学习机应该是全新的设计。

中级学习机使用的新技术笔者极力推荐 Z180cpu 和 6502cpu 结合。Z180 有较多的优点:Z180 的 DMA 可支持显示部件工作;Z180 的 mmu 好学好用,是分页的虚存结构;I/O 和中断多,便于系统设计;运行 CP/M 的内存扩展到 512K 以上;Z180 和 6502 的软件程序都可支持;双 cpu 的设计可发展矢量汉字,减小成本;保留原总线和时序,开发难度低;集成度提高,可提供更多的接口和功能;利用 Z180 串行接口可组成主从式网络。

中级学习机应是大键盘机箱结构,适合数字和五笔字形操作;有中华机总线并有更大的扩展空间;3.5 英寸软盘,提高操作系统性能;6502 和 Z180 结合使用可编程集成电路或门阵列技术;内存加大以便于数据库和电子表格等软件的运行;带网络接口和打印机接口等设备,便于用户扩展;利用 DMA 的特点,支持专用计算机显示器;实现高传真发声的动画功能,大力发展 CAI 教育,字库矢量化,以支持 24×24 点阵输出。

高级学习机针对大中专院校学生和专业人士的需求,与事务计算机兼容性好,适用于办公自动化等事务处理,其价格限制较小。本来 IBM 兼容机适合高级学习机,但近来计算机技术迅速发展 UNIX 操作系统,以太局域网,自动程序编写软件,多媒体技术也逐步纳入高级学习机的范围,要求高级学习机是工作站的雏形,就是显示解析度高且有灰度或彩色,扩展空间大且内存大,CPU 性能好且有一定开放程度。其中 CPU 性能最重要,虽说主板也可象扩展板一样替换,但国情使我们不能象国外计算机玩家那样想怎么换就怎么换,一般都是硬件一次投资,软件低价使的方式,笔者认为 80386SX CPU 较好。

高级学习机应是把公开版权的旧 UNIX 操作系统移植到高级学习机中,保留了 UNIX 各种特点,增加 80386 专有的虚拟 86 运行 DOS 操作系统,从而使软件运行的限制取消,使用工作站的单色带灰度显示,并在显示中解决电源问题,小机箱设计,80386SX CPU 有专用总线集成电路支持,其他类似 IBM-PC 系列兼容计算机。

三级学习机档次拉开很大,软件范围广,价格差别大,适应性很广。

笔者的设想仅是一个开始。在“复关”的时候制订受市场欢迎的标准对国内众多计算机厂商很有意义。

浅析 Turbo C 图像处理函数的缺陷 与像素的保存和恢复

屈景辉 西宁市西关大街 79 号自动化站(810008)

任京芳 西安市石油公司

Turbo C 提供的函数 `imagesize()`、`getimage()`、`putimage()` 和函数 `malloc()`、`farmalloc()`，前三者用来处理点阵图像，后二者为图像处理申请内存空间。在程序设计过程中发现这些函数在处理图像时有不足之处，深感遗憾。一些刊物也曾介绍过关于图形图像保存与恢复的文章，但未尽人意。笔者在实际工作中针对存在问题，以 VGA 图形模式 (12H) 为例，详细介绍视频 RAM 按位读写过程，并给出演示源程序。

一、问题分析

1. 函数 `imagesize()` 返回值为要求窗口 `x1, y1, x2, y2` 相应的内存大小，但这个图像所要求的字节不允许超过 64KB，若超过则返回 `0xffff(-1)` 表示出错。

2. 函数 `getimage()` 和 `putimage()` 要求的内存缓冲区 `buffer` 也不能超过 64KB。否则，图像的保存与恢复出现意外结果。

3. 若按图形模式要求重写 `imagesize()` 函数，计算所需内存字节数，用 `malloc()` 或 `farmalloc()` 申请内存，用 `getimage()`、`putimage()` 或者自行设计视频 RAM 直接读写仍然出错，即申请内存不可大于 64KB。

这样，对于超过 64KB 的图像处理，上述两种函数便显得无能为力。另外，有关刊物介绍的图形图像屏幕保存与恢复的文章大多数是将整屏信息保存在磁盘文件中，待需要时再从文件读出点阵信息，直接写入视频 RAM。此法优点是按字节处理，不占用内存，不受内存空间和文件大小限制；不足之处是整屏整字节读出写入，占用磁盘空间大，显示速度慢，灵活性差，不能按位(像素)处理。

二、功能要求及对策

在软件设计中，往往存在处理大于 64KB 图形图像的问题，使保存与恢复屏幕的范围在点与整屏 (0, 0, 639, 479) 之间变化，图形图像信息储存在磁盘文件或暂存 RAM 中。

1. 如果信息保存在磁盘文件中，为提高处理速度，可采用 MS-DOS 5.0 以上版本，在扩充或扩展内存区用 `RAMDRIVE.SYS` 文件开设虚拟盘，处理完毕将信息拷入物理硬盘。

在读盘写入时，为防止像素信息大于 64KB，可按位面申请空间，一次分块读出写入，避免边读边写速度慢或读一面写一面屏幕闪动的缺点。

2. 如果信息暂存 RAM 中，为克服 `malloc()` 函数的不足，可用中断调用的方法申请内存空间，或者用 `malloc()` 申请空间分块按位面存放。

三、实现方法

1. 图形图像处理区域字节计算：

在 VGA 图形模式 12H (640×480) 下，VGA 的 16 色是通过四个位平面表示的，屏幕上的一个像素在每个彩色位平面中各占 1 位，4 个位表示的像素允许同时显示 16 种颜色。4 个位平面占有相同的段地址，可以通过端口地址控制对某个位面的读写。在一个矩形区域中，X 起始和终止坐标存在下列三种情况：

(1) 在同一个字节内：

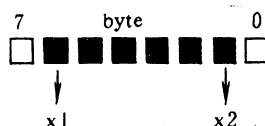


图 1

(2) 在相邻两个字节内：

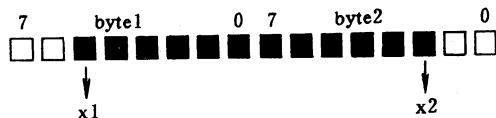


图 2

(3) 起始和终止坐标相隔若干个字节：

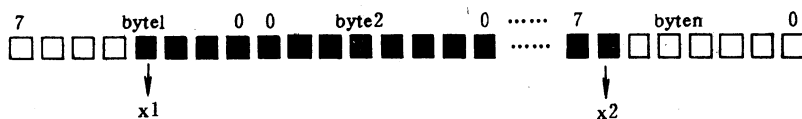


图 3

那么：

水平方向占用字节数： $p_byte = (endx / 8 - startx / 8) + 1$

垂直方向像素高： $i_high = endy - starty + 1$

X 起始坐标在字节中位置： $s_bit = 7 - startx \% 8$

X 终止坐标在字节中位置： $e_bit = 7 - endx \% 8$

矩形区域占用字节数：

$total_byte = p_byte \times i_high \times 4 + 8$

2. 屏幕像素位置与显示存储器中字节地址转换关系

系

在分辨率 640×480 模式下，同一位面的 8 个像素占一个字节，每行为 80 字节，每位面 38400 (640/8×

480)字节,屏幕像素与视频 RAM 地址映射关系为:

字节地址= $y \times 80 + x / 8$

位的位置($0-7$)= $7 - (x \% 8)$

3. 像素值的读出

根据图形控制器寄存器的 I/O 地址 3CE 指定(索引值为 4)读位面选择寄存器,读出位面通过 I/O 地址 3CF 指定($0-3$),读出字节数和地址根据上述方法确定。

4. 像素值的写入

由定序器索引寄存器 I/O 地址 3C4(索引值为 2)选择彩色位面写允许寄存器,由定序器数据寄存器 I/O 地址 3C5,指定写入位面。

在像素写入过程中,起始和终止位所在字节必须按位写入,二者之间按字节写入。如果按读出字节直接写,势必将多余位的值也一块写进去,造成要求保存区域与恢复区域不能精确对应的结果。因此,可分别按上述三种情况写入阴影部分。

当写入字节的某几位时,由 I/O 地址 3CE 指定屏蔽寄存器(索引值为 8)设定屏蔽位,然后将要写入地址中的内容读出,装入处理器锁存器(只要把内容读出即自动装入),再根据位屏蔽寄存器的设定写入一个字节的某几位,达到位屏蔽的目的。

四、程序设计

本例将屏幕内任一矩形区域保存在磁盘文件中,按位面分块一次读出写入,实现了按位、字节处理。保存与恢复函数可带参数文件名,能够在磁盘上保存多个图像文件随意恢复;能够将多个图像文件恢复在同一屏幕上拼接、叠加,弥补了 Turbo C 图象和申请内存函数的缺陷。

程序清单附后:

```
rest_image_file(int startx,int starty,char * file)
{ /* 图象文件写入视频 RAM */
char far *scr,far *buff[4];
int i_high,s_pix,e_pix,p_byte;
int j,plane,left=0xff,right=0xff,temp_y,temp;
long total_byte;
FILE *fp;
scr=(char far *)0xA0000000;
fp=fopen(file,"rb");/* 从文件中读出四个参数 */
fread(&i_high,2,1,fp);fread(&p_byte,2,1,fp);
fread(&s_pix,2,1,fp);fread(&e_pix,2,1,fp);
total_byte=p_byte * i_high;/* 每位面占字节数 */
for(j=0;j<=3;j++){
    buff[j]=(char far *)malloc(total_byte);
    fread(buff[j],total_byte,1,fp);
    /* 为位面分配缓冲区,并从文件中读入 */
    left=left<<s_pix+1; /* 计算写入位 */
    right=right>>8-e_pix;
    outportb(0x3c4,2); /* 设定写入位面 */
    outportb(0x3ce,8); /* 设定位屏蔽 */
```

```
temp_y=starty+i_high>479? 479-i_high:starty;
switch(p_byte)
{
case 1; /* 一字节处理 */
    outportb(0x3cf,~(left | right));/* 设定屏蔽位 */
    for(plane=0; plane<=3; plane++){
        outportb(0x3c5,0x01<<plane); /* 写位面选择 */
        for(j=temp_y;j<temp_y+i_high;j++){
            temp=(scr+startx/8+j*80);
            /* 写入地址内容装入锁存器 */
            memcpy((scr+startx/8+j*80),\
                (buff[plane]+(j-temp_y)*p_byte),p_byte);
        }
        break;
case2: /* 二字节处理 */
    for(plane=0;plane<=3;plane++){
        outportb(0x3c5,0x01<<plane);
        for(j=temp_y;j<temp_y+i_high;j++){
            outportb(0x3cf,~left);
            temp=(scr+startx/8+j*80);
            memcpy((scr+startx/8+j*80),\
                (buff[plane]+(j-temp_y)*p_byte),1);
            outportb(0x3cf,~right);
            temp=(scr+startx/8+j*80+1);
            memcpy((scr+startx/8+j*80+1),\
                (buff[plane]+(j-temp_y)*p_byte+1),1);
        }
        break;
default: /* 多字节处理 */
    for(plane=0;plane<=3; plane++){
        outportb(0x3c5,0x01<<plane);
        for(j=temp_y;j<temp_y+i_high;j++){
            outportb(0x3cf,~left);
            temp=(scr+startx/8+j*80);
            memcpy((scr+startx/8+j*80),\
                (buff[plane]+(j-temp_y)*p_byte),1);
            outportb(0x3cf,~0xff);
            memcpy((scr+startx/8+j*80+1),\
                (buff[plane]+(j-temp_y)*p_byte+1),
                p_byte-2);
            outportb(0x3cf,~right);
            temp=(scr+startx/8+j*80+p_byte-1);
            memcpy((scr+startx/8+j*80+p_byte-1),\
                (buff[plane]+(j-temp_y)*p_byte+p_byte-1),1);
        }
        break;
    }
}
```

```

outportb(0x3c4,2);      outportb(0x3ce,8);
for(j=0;j<=3;j++) free(buff[j]);
fclose(fp);return 0;
}
/* 像素的保存与恢复演示源程序清单 */
#include <graphics.h>
#include <alloc.h>
#include <string.h>
#include <stdlib.h>
#include <stdio.h>
#include <dos.h>
main()
{
int gd=DETECT,gm;
initgraph(&gd,&gm,"");
setbkcolor(1);
setfillstyle(1,GREEN);
bar(40,30,600,380);
setfillstyle(7,LIGHTRED);
bar(32,22,590,370);
save_image_file(0,5,639,400,"QJH-SCR.PIC");
getch();
clearviewport();
rest_image_file(0,50,"QJH-SCR.PIC");
getch();
closegraph();
}

```

```

}
save_image_file(int startx,int starty,int endx,int
endy,char * file)
{ /* 矩形区域屏幕图像存入磁盘文件 */
char far * scr;
int i-high,s-pix,e-pix,p-byte,j,plane;
FILE * fp;
i-high=endy-starty+1; /* 像素高度 */
p-byte=(endx/8-startx/8)+1; /* 像素占用字节 */
s-pix=7-startx % 8; /* 像素起始位 */
e-pix=7-endx % 8; /* 像素终止位 */
scr=(char far *)0xA0000000;
fp=fopen(file,"wb"); /* 四个参数写入文件 */
fwrite(&i-high,2,1,fp); fwrite(&p-byte,2,1,fp);
fwrite(&s-pix,2,1,fp); fwrite(&e-pix,2,1,fp);
outportb(0x3ce,4); /* 设定读位面 */
for(plane=0;plane<=3;plane++) {
outportb(0x3cf,plane); /* 按位面 0-3 读出 */
for(j=starty;j<starty+i-high;j++)
fwrite((scr+j*(640/8)+startx/8),p-byte,1,
fp);
} /* 写盘 */
fclose(fp);
outportb(0x3ce,3); outportb(0x3cf,0);
}

```

一个多功能的硬盘主引导程序

燕士海 南京市邮政科研所(210010)

只要懂操作系统的读者都会知道,硬盘在启动过程中,要执行主引导记录程序,其主要功能有三个:一是检查硬盘分区的正确性,二是检查 DOS 活动分区是否存在两个系统文件,三是释放引导权给 BOOT 记录。

本人自编一个新的主引导记录程序,除了保持上述全部功能外,还具备以下三个特殊功能。

1. 硬盘引导成功后,利用时钟中断(18.2 次/秒),对 INT 21H 向量进行监察,一旦该向量被修改,就立即蜂鸣报警,然后再修改正确,该功能将对一些以 INT 21H 中断作为传播途径的文件型病毒起到很好的免疫作用。

2. 软盘启动后进不了硬盘,该功能对于通过软盘启动后将病毒传给硬盘的系统型病毒来说,是极好的防治。

3. 硬盘在启动过程中,需用户输入 PASSWORD,这个功能对于 286 以上的微机而言已算不上特殊了,

但对于无 CMOS 设置的 PC 机来说是比较实用的。

该主引导记录的编程思想是利用硬盘启动过程中必须执行 0 头 0 柱 1 扇区内容的特点,将原主引导记录从 0 头 0 柱 1 扇区搬到 0 头 0 柱 3 扇区,新的主引导记录保存在 0 头 0 柱 1 扇区,这样就保证了硬盘在设备自检完成后先执行新主引导程序,再执行原主引导程序,直至引导成功。

新主引导程序清单见附页。

该主引导程序之所以能实现上述特殊功能,主要采取了三个动作。

1. 通过修改时钟中断 INT 8(18.2 次/秒)中断向量使之指向一段用户程序,程序中对 INT 21H 的向量进行判别是否被修改了,然后再跳转到原来的 INT 8 中断入口。

2. 隐藏了硬盘介质标志,软驱启动后因无法找到磁盘介质标志而不承认硬盘的存在。

3. 通过修改 INT 13H 中断向量,使之也指向一段

用户程序,该程序中将说明磁盘介质的存储位置,从而使得硬盘能正确访问其主引导记录。

需要说明的是,新的 INT 8 和 INT 13H 部分内容必须驻留内存。本程序中的解决方案是将它们推向内存高端 9000:0 处(针对于 640K 常规内存而言最大不超过 9F00:0,如果 512K 常规内存可为 7500:0,最大不得超过 7F00:0),以保证 DOS 在正常情况下访问不到这段内存,如果读者要稳妥起见,也可以将这段内存改为隐含,以不被 DOS 利用。此外程序中的部分源指针值(SI)和目标指针(DI)是根据代码偏移地址值而定的,它们将随程序偏移地址的变化而变化,各位读者务请注意。

本程序中 PASSWORD 为 911,也可以任意改为其它三字节数或字母,这段程序已在数台不同型号的机子上调试通过,实践证明它能有效地防止某些文件型病毒的传染和破坏,如果有读者需要本程序的详细说明及安装过程,可直接与本人联系(TEL:025—4485381)。

新主引导程序清单:

CS:0100	CLI	
CS:0101	SUB	AX,AX
CS:0103	MOV	SS,AX
CS:0105	MOV	ES,AX
CS:0107	MOV	DS,AX
CS:0109	MOV	AX,7C00
CS:010C	MOV	SP,AX
CS:010E	STI	
CS:010F	MOV	SI,AX ;代码下移 200H
CS:0111	MOV	DI,7E00
CS:0114	CLD	
CS:0115	MOV	CX,0100
CS:0118	REPZ	
CS:0119	MOVSW	
CS:011A	JMP	031D
CS:011D	MOV	AX,0201 ;读原主引导记录
CS:0120	MOV	CX,0003
CS:0123	MOV	DX,0080
CS:0126	MOV	BX,7C00
CS:0129	INT	13
CS:012B	NOP	
CS:012C	NOP	
CS:012D	NOP	
CS:012E	MOV	DI,0002 ;二次机会输
CS:0131	MOV	SI,7F0C ;入 PASSWORD
CS:0134	MOV	CX,000C
CS:0137	LODSB	
CS:0138	MOV	BX,0007
CS:013B	MOV	AH,0E
CS:013D	INT	10
CS:013F	LOOP	0137 ;显示 'PASSWORD'
CS:0141	MOV	SI,7F18
CS:0144	MOV	BP,0003
CS:0147	MOV	AH,00
CS:0149	INT	16
CS:014B	CMP	[SI],AL ;判断 PASSWORD
CS:014D	JNZ	0155
CS:014F	INC	SI
CS:0150	DEC	BP
CS:0151	JNZ	0147
CS:0153	JMP	0163
CS:0155	MOV	BX,0007
CS:0158	MOV	AH,0E
CS:015A	MOV	AL,58
CS:015C	INT	10
CS:015E	DEC	DI
CS:015F	JNZ	0131
CS:0161	JMP	0161 ;PASSWORD 出错,死机
CS:0163	MOV	SI,004C ;填充 INT13 跳转入口
CS:0166	MOV	DI,7EC7
CS:0169	LODSW	
CS:016A	STOSW	
CS:016B	LODSW	
CS:016C	STOSW	
CS:016D	MOV	SI,0020 ;填充 INT8 跳转入口
CS:0170	MOV	DI,7EFD
CS:0173	LODSW	
CS:0174	STOSW	
CS:0175	LODSW	
CS:0176	STOSW	
CS:0177	MOV	DI,000 ;将 INT13 和 INT8 增添
CS:017A	MOV	SI,7EB 部分推向内存高端
CS:017D	MOV	CX,0052
CS:0180	MOV	AL,[SI]
CS:0182	PUSH	DS
CS:0183	PUSH	AX
DS:0184	MOV	AX,9000
CS:0187	MOV	DS,AX
CS:0189	POP	AX
CS:018A	MOV	[DI],AL
CS:018C	POP	DS
CS:018D	INC	DI
CS:018E	INC	SI
CS:018F	LOOP	0180
CS:0191	MOV	BX,004C ;修改 INT13 向量
CS:0194	MOV	AX,0000
CS:0197	MOV	[BX],AX
CS:0199	MOV	AX,9000
CS:019C	MOV	[BX+02],AX
CS:019F	MOV	AX,001A
CS:01A2	MOV	BX,0020 ;修改 INT8 向量
CS:01A5	MOV	[BX],AX
CS:01A7	MOV	AX,9000


```

CS:01AA  MOV    [BX+02],AX
CS:01AD  JMP     0201
CS:01AF  NOP
CS:01B0  NOP
CS:01B1  PUSHF   ,INT13 补充内容
CS:01B2  CMP     DX,0080
CS:01B6  JNZ     01C5
CS:01B8  CMP     CX,+01
CS:01BB  JNZ     01C5
CS:01BD  CMP     AH,02
CS:01C0  JNZ     01C5
CS:01C2  MOV     CX,0003
CS:01C5  POPF    ,跳转到 INT13 入口
CS:01C6  JMP     0000:0000 ,地址待填
CS:01CB  PUSHF   ,INT8 补充内容
CS:01CC  PUSH    DS
CS:01CD  PUSH    SI
CS:01CE  PUSH    DI
CS:01CF  PUSH    AX
CS:01D0  XOR     AX,AX
CS:01D2  MOV     DS,AX
CS:01D4  PUSH    ES
CS:01D5  MOV     ES,AX
CS:01D7  MOV     SI,0084
CS:01DA  LODSW
CS:01DB  CMP     AX,1460;判断 INT21 偏移地址
CS:01DE  JNZ     01E6
CS:01E0  LODSW
CS:01E1  CMP     AX,0270 ,判断 INT21 段地址
CS:01E4  JZ      01F6
CS:01E6  MOV     AX,0E07 ,报警
CS:01E9  INT     10
CS:01EB  MOV     DI,0084
CS:01EE  MOV     AX,1460
CS:01F1  STOSW
CS:01F2  MOV     AX,0270
CS:01F5  STOSW ,重新填充 INT21 向量
CS:01F6  POP     ES
CS:01F7  POP     AX
CS:01F8  POP     DI
CS:01F9  POP     SI
CS:01FA  POP     DS
CS:01FB  POPF    ,跳转到 INT8 入口
CS:01FC  JMP     000,0000 ,地址待填
CS:0201  MOV     AX,0E0D
CS:0204  INT     10
CS:0206  JMP     0000:7C00 ,跳转到原主引导记录

```

新主引导记录内存映像:

```

CS: 020C 0A 0D 50 41
CS: 0210 53 53 57 4F 52 44 20 20-39 31 31 0E CD B9 64 65

```

```

CS:0220 1F EB 93 00 00 00 00-00-00 00 00 00 00 00 00
.....
CS:02B0 00 00 00 00 00 00 00-00 00 00 00 00 80 01
CS:02C0 01 00 00 04 D1 02 11 00-00 00 EE FF 00 00 00 00
CS:02D0 C1 03 05 04 D1 CF FF FF-00 00 11 44 00 00 00 00
CS:02E0 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00
CS:02F0 00 00 00 00 00 00 00-00 00 00 00 00 00 55 AA

```

四通 MS-2401 与北大方正 排版文件的转换

王景余 57318 部队 10 中队

四通打字机在国内办公自动化领域应用较为广泛,但四通机编辑和排版所形成的文件却是非文本文件,它所编辑或排版的文件只能在同类型机器上查阅或使用,而与目前普遍使用的 286、386 等机器不能兼容,给用户带来了极大不便。本人用 Borland C++ 编写了其阅读程序,并做了 MS-2401 与北大方正排版文件互换的尝试,取得了成功。此程序在 286 和 386 微机上,都已通过并投入使用,效果良好。现将部分程序附后。

(1)HGTOST.C 是用于方正文本文件转四通 MS-2401 文本文件。

(2)STTOHG.C 是用于四通 MS-2401 非文本文件转方正文本文件。

(1)/ * hgtost.c 方正文本文件转四通 MS-2401

```

文件 * /
#include<stdio.h>
main()
{
char c,name1[80],name2[80];
FILE *fp,*fp1,*fopen();
printf("\r\n\n          请输入方正源文件名:
");
scanf("%s",name1);
printf("\r\n          请输入四通目标文件名:
");
scanf("%s",name2);
if((fp=fopen(name1,"rb"))==0)
{
printf("\r\n\n          * * * * * '%s'",name1);
printf("不存在,请重新启动! * * * * * \r\n");
fclose(fp);
exit(0);
}
}

```

```

fp1=fopen(name2,"wb");
while((c=getc(fp))!=EOF)
if((c&0x80)==0x80)
{putc(c,fp1);
putc((getc(fp)&0x7f,fp1);
}
else putc(c,fp1);
printf("\r\n\n      四通目标文件");
printf("%s"已经形成,请君使用该文件! \r\n\n",
name2);
return;
}
(2)/ * sttoh.c 四通 MS-2401 非文本文件转方正文
本文件 */
#include<stdio.h>
main()
{
char name1[80],name2[80]
int c,d;
FILE *fp1,*fp,*fopen();
printf("\r\n\n      请输入四通源文件名: ");
scanf("%s",name1);
printf("\r\n      请输入方正目标文件名:");
scanf("%S",name2);
if((fp=fopen(name1,"rb"))==0)
{
printf("\r\n\n      * * * * * '%s'",name1);

```

```

printf("不存在,请重新启动! * * * * * \r\n");
fclose(fp);
exit(0);
}
fp1=fopen(name2,"wb");
while((c=getc(fp))!=EOF)
{
d=c&0xff;
if(d>=0x80&&d<0xa0)
{
c=getc(fp);
d=c&0xff;
continue;
}
else if(d>=0xa1)
{
putc(d,fp1);
putc(getc(fp)|0x80,fp1);
}
else putc(c,fp1);
}
printf("\r\n\n      方正目标文件 '%s'",name2);
printf("已经形成,请君使用该文件! \r\n\n");
return(0);
}

```

谈谈自由组合条件查询语句的形成

庄志 广州广东商检局

在数据处理程序中,免不了要对数据库进行查询、修改、删除、统计、打印等操作,而要实现这些目的又离不开查询条件语句。因此查询条件的形成在整个程序中起着重要的作用。查询条件的组合形式,通常可分为固定组合、选择组合和自由组合。其中自由组合条件查询的形式应用最灵活,可适用于各种不能在编程时预先确定的统计项目及模糊查询等要求。本文就笔者的实践体会举例(FoxBASE 语言)谈谈自由组合条件查询语句的形成。

自由组合,关键就是能让程序使用者自由地从各种条件中组合成其需要的查询条件。所以首先要把所有可能的条件都列出来供使用者选择,然后再根据选择的结果逐步组合成查询语句。

如现有一个工厂登记管理的数据库 GCDJ.DBF,其结构如下:

字段名	类型	长度	小数位数
编号	C	10	—
产品类别	C	8	—
厂名	C	20	—
厂址	C	20	—
电话	C	7	—
联系人	C	8	—

编一段子程序 LNA.PRG 把所有的字段名都列于屏幕上供选择:

```

.type LNA.PRG
parameters cx1
clear
@3,20 say "1. 编号"
@3,40 say "2. 产品类别"
@4,20 say "3. 厂名"
@4,40 say "4. 厂址"

```

```
@5,20 say "5. 电话"
@5,40 say "6. 联系人"
store space(20) to cx1
@6,16 say "请选择查询条件项目,各项目间以逗号相隔。"
@7,16 get cx1
read
return
```

在这个例子中各条件间的组合关系是逻辑“与”的关系。如既需要“与”的关系,又需要“或”和“非”的关系。那么在列字段名时应把各种逻辑关系也一同列出来供选择。

当 LNA. PRG 运行后,使用者可根据需要选出查询条件的项目。如现需要查询某间名为“……东风……厂”的服装厂,可选 2,3,3,也就是 cx1=“2,3,3”。

cx1 是查询条件的基本结构,有了这个结构后,下一步就要对每一项输入具体的数值了。

```
. type CXLZ. PRG
set talk off
set echo off
set step off
do while .t.
store " " to cx1,cx6,cx7,cx8
use GCDJ. DBF
do LNA. PRG with cx1
cx2=cx1
do while .t.
clear
cx3=val(cx2)
if cx3<1 .or. cx3>6
exit
endif
cx2=substr(cx2,3)
@3,10 say "请输入查询条件的具体数值。"
cx4=field(cx3)
@5,10 say cx4+";"
cx5=len(&cx4)
cx5=space(cx5)
@5,20 get cx5
```

```
read
cx5=trim(cx5)
cx6=cx6+';'+'+cx4+'含'"+cx5+"'"
cx7=cx7+'.and.'+'+'"+cx5+'"'$'+cx4
enddo
cx6=substr(cx6,4)
cx7=substr(cx7,7)
@4,18,say "查询条件为:"
@6,22 say cx6
@8,18 say "以上条件正确吗(Y/N)?" get cx8
read
if upper(cx8)="Y"
exit
endif
enddo
clear
@3,10 say "cx6=="+"cx6
@5,10 say "cx7=="+"cx7
return
```

上面 CXLZ. PRG 运行后便可得出查询条件 cx7。有了查询条件后便可进行各种操作了。

如: loca all for &cx7 查询
 count for &cx7 to n 统计
 set filter to &cx7 过滤

在前面查服装厂的例子中,只要依次输入“服装”、“东风”和“厂”便能得出查询条件:

“服装”\$ 产品类别.and.“东风”\$ 厂名.and.“厂”\$ 厂名。

CXLZ. PRG 只是一个示范例子,在实际中情况会更复杂一些。如要考虑容错问题、当字段中有数值类型和日期类型时该如何处理等。当然 CXLZ. PRG 是核心。而 CXLZ. PRG 的指导思想是把查询条件逐一地组合成一个完整的查询语句。

从上面的讨论示例可见用自由组合条件查询语句可完成各种综合统计和模糊查询的功能。但要求使用者必须是具备一定的逻辑代数知识,对于具有“与”、“或”、“非”功能的自由组合条件则更是如此。

Keyboard 命令在全屏编辑中的使用

沙 玮 四川甘洛县财政局微机室(616850)

在 FoxBASE(2.00 或 2.10)中有一个常被忽略的命令——Keyboard,在大多数 FoXBASE 编程实例中都没有涉及,但利用它编写出的二维数组全屏编辑程序,可大大优化 FoxBASE 程序的编辑界面。Keyboard 的功能是向键盘缓冲区中塞入字符串,供后面的交互式命令使用。通常在输入编辑多个字段、变量或一个数

组时,使用多个格式输入命令@row,col get[表达式]接一个 read 的方法,这样光标虽可在各变量之间移动,但只能以 get 命令的顺序转移,若要实现全屏编辑二维数组变量的光标移动功能(→、↑、←、↓、用[PgUp][PgDn]、^→、^←翻页)只能用 Inkey()截获的键值在新的位置 Get 另一个变量。但这样录入的数

据的第一个 ASCII 码会因为 Inkey() 的调用而丢失, 解决的办法就是用 Keyboard Chr(Inkey()) 将丢失的键值输入键盘缓冲区后再执行 Read, 执行结果和通常的 Get—Read 一样, 但可根据光标、功能控制键进行其它动作。另外在选择新的编辑变量时, 必须关闭原来 Get 的变量, 最方便的办法就是 Keyboard Chr(13) 再执行 Read。笔者编写的工资管理程序的编辑模块, 是把工资库中的数据用 Scatter 传送给一个二维数组, 再用上述方法进行编辑。采用数组作中介的好处是在误操作时不会破坏数据库, 并且可方便地进行各种计算。在编辑过程中根据用 Inkey() 截获的键值完成预先定义的各种操作或子程序, 如: 空回车后横向或纵向转移编辑光标、定义 F3 键为拷贝上行数值并回车、F5 计算合计、F6 键调用打印子程序等……。因此在使用时有种表格软件的感觉, 并可方便地写入扩充的功能。

附: 程序概要

```
DIME A(X,Y)
数组赋初值
X=1
DO WHILE X<=X1 && 将数组值显示出来
Y=1
DO WHILE Y<=Y1
@4+Y,3+10*X SAY A(X,Y)
Y=Y+1
ENDD
X=X+1
```

(上接 24 页)

的 ASCII 字符码, 而生成“扩展码”, 扩展码的长度也为 2 字节; 低字节必为 0 (扩展码标志), 高字节是相应代码, 如: (0;59) 代表功能键 F1, 常用的扩展码键如图 2—4。

键	高字节 代码值	键	高字节 代码值
F1—F10 键	58—68	Home	71
上 箭 头 键	72	End	79
左 箭 头 键	75	Ins	82
右 箭 头 键	77	Del	83
下 箭 头 键	80	Ctrl Prtsc	114

图 2—4 常用的扩展码键

(3) 如果按下的是换档键、双态键或控制键, INT 9H 不将键值送缓冲区, 而将键的状态送入内存 40:17H 和 40:18H 两个字节中, 这两个字节各位的含义如图 2—5, 用程序改变有效位中的一个, 就等效于实际敲了对应的键。

```
ENDD
X=1
Y=1
DO WHILE .T. && 开始编辑
@4+Y,3+10*X GET A(X,Y)
K=0
DO WHILE K=0 && 读入键值
K=INKEY()
ENDD
DO CASE && 判断键值
CASE K=24 && 键入向下键的处理
Y=Y+1
KEYBOARD CHR(13) && 关闭原来 GET 的变量
READ
LOOP
CASE K=..... && 处理其它控制键
:
:
:
CASE K>46 . AND. K<58 && 键入数字键的处理
KEYBOARD CHR(K) && 将键值塞入键盘缓冲区
READ && 继续输入新的数据
Y=Y+1 && 光标下移
LOOP
ENDC
ENDD
```

位	键	该位为 1 时的含义
7	Insert	插入态
6	Caps Lock	大写态
5	Num Lock	数字态
4	Scroll Lock	该状态开
3	Alt	该键按下
2	Ctrl	该键按下
1	左 Shift	该键按下
0	右 Shift	该键按下

图 2—5a 内存 40:17H 单元各位的含义

位	键	该位为 1 时的含义
7	Insert	该键按下
6	Caps Lock	该键按下
5	Num Lock	该键按下
4	Scroll Lock	该键按下

图 2—5b 内存 40:18H 单元各位的含义

一种汉字无级放大的方法

陈德运 李秀忠 哈尔滨科技大学计算机系 915 信箱(150080)

通常在西文 DOS 状态下仅能显示西文字符。在这篇文章里介绍在西文 DOS 状态下显示及放大显示汉字的方法。该方法是直接从标准汉字库(CCLIB.DAT)中提取汉字字模,然后利用一个字模放大算法重新构造汉字字模的点阵,再把它显示到屏幕上。下面我们详细地介绍这种方法。

1. 汉字库的结构及字模的提取

在汉字信息处理系统中,为了显示或打印汉字,必须建立汉字库,汉字库用来存储汉字的字形信息。目前汉字的字形信息普遍采用点阵形式进行存储,常用的点阵有 16×16 , 24×24 , 32×32 点阵库。我们使用的是 16×16 点阵汉字库,每个汉字在 16×16 点阵汉字库中是用 16 个整型数表示,存放在 32 个字节里,一个整数定义汉字点阵的一行,用代码的形式存放在两个字节里,低字节存放汉字每一点阵行的前 8 位,高字节存放汉字点阵行的高 8 位。一个汉字由两个机内码表示。假定某汉字“与”掉最高位的两个机内码值为 P1 和 P2,那么该汉字在汉字库中的位置可使用下面公式得到:

$$\text{位置} = ((P1 - 33) \times 94 + P2 - 33) \times 32L$$

有了上面计算汉字在汉字库中位置公式,就可利用此公式计算任意汉字在字库中的位置并能编程提取出它的字模,下面是提取汉字字模的过程。

```
get-mode(char *p, int *mode) /* 提取汉字字模 */
{
    FILE *fp;
    int i, j;
    long len;
    if(! (fp=fopen("cclib.dat", "rb"))){
        printf("File cclib.dat can not be opened");
        exit(0);
    }
    i=p[0]&127;
    j=p[1]&127;
    len=((i-33)*94+j-33)*32L;
    fseek(fp, len, 0);
    fread(mode, 1, 16, fp);
    fclose(fp);
}
```

其中 p[0], p[1] 存放的是要显示汉字的低位字节和高位字节的机内码,从字库中取出的汉字字模放在由 mode 所指的整型指针中,长度为 16。

2. 汉字字模的无级放大算法及函数

设字模由 16×16 矩阵表示,元素为“1”表示亮,“0”表示暗,则每个汉字字模矩阵如下:

$$\begin{bmatrix} a_{1,1} & a_{1,2} & \dots & a_{1,16} \\ a_{2,1} & a_{2,2} & \dots & a_{2,16} \\ \vdots & \vdots & & \vdots \\ a_{16,1} & a_{16,2} & \dots & a_{16,16} \end{bmatrix} \quad \text{其中 } a_{i,j} = \begin{cases} 0 & \text{暗} \\ 1 & \text{亮} \end{cases}$$

如果要放大某一汉字,并且给出它的放大比例,那么就可把字模矩阵上的某一点 $a_{i,j}$ 按此比例转换成一个点阵,放大算法如下:

假定横向和纵向放大比例分别是 x_l 和 y_l ,那么对应此汉字字模上的每一点 $a_{i,j}$ ($i=1 \dots 16, j=1 \dots 16$) 放大后的矩阵为:

$$A_{i,j} = \begin{bmatrix} b_{1,1} & b_{1,2} & \dots & b_{1,n} \\ b_{2,1} & b_{2,2} & \dots & b_{2,n} \\ \vdots & \vdots & & \vdots \\ b_{m,1} & b_{m,2} & \dots & b_{m,n} \end{bmatrix}$$

其中 $b_{i,j} = a_{i,j}$, $n = i \times x_l, y = j \times y_l$ 。假设原汉字点阵的显示位置左上角坐标为 (x_0, y_0) ,则放大后 $a_{i,j}$ 的放

大点阵显示位置为 (x_1, y_1) , $x_1 = \sum_{n=0}^{i-1} n \times x_l, y_1 = \sum_{n=0}^{j-1} n \times y_l$ 。具体的放大过程如下:

```
show-mode(mode, x, y, x_l, y_l, color, bkcolor) /* 汉字无级放大 */
int mode[16], x, y, color, bkcolor;
float x_l, y_l;
{
    int i, j, x0, y0, x1, y1, n, h, xx, yy, k, x, c;
    y1 = y; y0 = 0;
    for(j=0; j<16; j++){
        x1 = x; x0 = 0; k = 128;
        for(i=0; i<16; i++){
            if(i%8) k = 128;
            s = mode[j]&k;
            k >>= 1;
            if(s != 0)
                c = color;
            else c = bkcolor;
            xx = i * x_l; yy = j * y_l
            for(n=0; n<yy; n++){
                for(h=0; h<xx; h++){
                    putpixel(x1+h, y1+n, c);
                    x += xx - x0
                    x0 = xx
                }
                y += yy - y0;
                y0 = yy
            }
        }
    }
```

其中 mode 为存放汉字字模的整型数组,长度为 16。(x,y)为汉字显示位置,xl,yl 为横纵向放大比例。color 为汉字显示颜色,bkcolor 为显示背景颜色。

3. 结束语

以上介绍的方法已在 Compaq/33M 机上用 Tur-

bo C 语言实现,使用此方法在放大比例较大及控制步长较大时被放大的汉字有锯齿状。通过适当修改放大点阵矩阵即可消除锯齿。修改放大点阵矩阵的方法我们将另文发表。

倍受用户青睐的 DR DOS 6.0

李瑛彬 兰州军区司令部作战部

提及 DR DOS 人们似乎感到很陌生,难怪于 MS DOS 和 PC DOS 垄断整个 PC 系列计算机市场。DR DOS 6.0 实际上是美国 Digital Research 公司在 MS-DOS 5.0 基础上推出的新型操作系统。我们从系统的启动信息(Copyright(c)1976,1982,1988,1990,1991, Digital Research Inc.)中可以看出,DR DOS 的发展也经过了近 20 年的旅程。该系统刚在市场出现就受到广大计算机工作者的青睐,因为它不仅完全兼容了 PC DOS 和 MS DOS 功能,更重要的是拓展了系统的强大功能,使 DOS 日趋完善,用户使用得心应手。DR DOS 具有压缩存取文件、多任务切换、高级内存管理、优化磁盘、加速、抵御病毒攻击、口令保护以及图形界面等特点。它可以将自己置于高端内存区,而把更多的常规内存让给用户使用。总之,DR DOS 的特点很多,为便于读者理解和使用,下面加以简要介绍。

一、学习和帮助

DR DOS 6.0 为用户提供了全方位服务帮助功能,可用两种方法获得帮助信息。

第一种是:

C>help 命令名(内部命令或外部命令)✓ 或
C>dosbook 命令名(内部命令或外部命令)✓
如:C>dosbook chkdsk✓

第二种是:

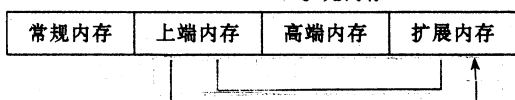
C>命令名/H 或/?
如:C>xdir/? 或 C>xdir/h

这两种帮助信息中 DOSBOOK 的内容较为详细,/? 或/H 的信息较为简洁,但使用比较方便。

二、内存管理

从 MS DOS 5.0 开始就对内存实施有计划的管理和使用,以减少过多的内存(主要指上端内存、高端内存、扩展内存等)被浪费掉。因 DOS 5.0 对中文系统的支持不尽完美,故计算机用户对其也不是那么乐观。

0KB 640KB 1MB → 扩充内存



科学地使用内存,可节省大量内存空间。如运行 WBZX.EXE 程序,若将该程序装入上端内存,可节省内存 43K,但必须对系统文件进行明确的重新设置。

C>type config.sys
hidos=on ;将系统的数据结构移到上端内存区
break=on ;允许程序在运行时用 Ctrl+C 或 Ctrl+Break 中断

files=25 ;可打开文件个数
hibuffers=20 ;在高端内存开设 20 个缓冲区
hidevice=c:\dos\emm386.sys; 将 386 以上微机驱动程序安装到上端内存

这时的内存为 586K,再运行 WBZX.EXE 程序,就变为 389K,若在运行 WBZX.EXE 时按下面方法:

C>hiloal wbzx✓;将 WBZX 安装到高端内存
内存可剩 432K,仅仅一条命令就可节省内存 43K。

三、文件和目录管理

DR DOS 6.0 与其它低版本的 DOS 系统相比,在文件和目录的管理上比较完善和安全。

在文件管理方面:

对文件进行删除时设置了安全参数供用户参考,以决定是否要真正删除文件。如要删除扩展名为 BAK 的所有文件,在后面加一个安全缓冲参数以提醒用户是否真正删除,如删除回答 Y,否则为 N:

C>del *.bak/c✓
94WZ1.BAK(Y/N)? Y
94WZ2.BAK(Y/N)? N

XDEL,ERAQ,ERASE 命令加上参数,也可以事先征求用户意见是否删除文件。

除此之外,对删除后的文件可以用 UNDELETE 命令予以恢复。

在目录管理方面:

DIR,XDIR 命令增加了很多显示磁盘目录的综合管理信息。RENDIR 命令可以更换子目录名。MOVE 则可以把文件从一个子目录转移到另一个子目录中。

另外,为防止别人删除、拷贝、阅读你的文件或使用你的子目录,PASSWORD 命令则对文件和子目录设置口令,以达到保密的目的。

四、命令简介

DR DOS 6.0 除兼容了 MS DOS 3.31 以下版本的命令外,还扩充了许多新的命令,由于篇幅有限,只对新增加的内外部命令加以简要介绍。无论对兼容的或者是新增加的命令,具体的操作方法读者可以在使

用期间在命令后加入/H或/?求得系统较为详细的提示与帮助。

(一)内部命令

1. 命令集

BREAK CHCP CHDIR/CD CLS COPY
CTTY DATE DEL DELQ DIR ERAQ
ERASE/ERA EXIT HILOAD MKDIR/MD
MORE PATH PROMPT RENAME/REN
RMDIR/RD SET TIME TYPE VER VERI-
FY VOL.

2. 新增命令的使用

C>DELQ 文件及通配符说明 [/S]✓

这条命令可以删除包含系统文件(加参数/S)在内的文件,但在删除之前预先询问,如果删除键入Y,否则为N。下例删除file1.bak,不删file2.bak:

C>delq *.bak✓

FILE1.BAK(Y/N)y

FILE2.BAK(Y/N)n

C>ERAQ✓

方法与DELQ命令相同。

C>EXIT✓

可以返回到另一运行的程序或DR DOS。

(二)外部命令

DR DOS 6.0的外部命令可以在低版本的DOS环境下运行,且用DIR命令可以显示出来。

1. 命令集

APPEND ASSIGN ATTRIB BACKUP
CHKDSK COMMAND COMP CURSOR
DELPURGE DELWATCH DISKCOMP
DISKCOPY DISKMAP DISKOPT DOSBOOK
EDITOR EXE2BIN FASTOPEN FC FDISK
FILELINK FIND FORMAT GRAFTABL
GRAPHICS JOIN KEYB LABEL LOCK
MEM MEMMAX MODE MOVE NLSFUNC
PASSWORD PRINT RECOVER RENDIR
REPLACE RESTORE SETUP SHARE SID
SORT SUBST SUPERPCK SYS TASKMAX
TOUCH TREE UNDELETE UNFORMAT
UNINSTAL VIEWMAX XCOPY XDEL
XDIR.

2. 新增命令的使用

C>CURSOR [Sn][C][off]✓

这是一个修改光标及其闪烁频率命令。Sn是两位十进制数值,用它来设置闪烁间隔,闪烁间隔为n乘以1/20秒,其使用范围是1~20,缺省值为一秒的4/20;/C可以允许CGA兼容,/off可终止软件生成光标,并恢复硬件光标。

C>DELPURGE 文件及通配符说明 [/A][L][S][P][D:date][T:time]✓

该命令是释放未决删除文件所占的磁盘空间,主要指DELWATCH命令保存的未决删除文件。未决删

除文件,是指当某些文件被删除后在列目录时显示不出,实际上仍保留在磁盘上,可在任何时候用UNDELETE命令恢复的文件。为避免未决删除文件占去大量的磁盘空间,可用DELPURGE命令和DELWATCH命令对其进行限定控制,如它的最大文件限额可以是1至65535。/A删除文件时不提示,缺省时提示;/L列出指定的文件,但不删除;/S抹除指定子目录下所有子目录中的文件;/P显示满一屏时暂停;/D:date或/D:-nn抹除设定日期或设定一个天数范围的文件;/T:time抹除指定时间前删除的文件。

C>DELWATCH 文件及通配符说明 [/S][D][B:n][O:ext...][E:ext...][F:n][MU][ML]✓

保存磁盘上注销文件作为未决删除文件。/S显示该命令当前状态;/D中止指定盘中该命令;/B:n指定存储的同名和同路径文件数量,n范围为1至65535,缺省值为1;/O:ext仅存储指定的扩展名文件;/E:ext仅存储指定扩展名以外文件;/F:n设定要存储文件的最大值,缺省值为200个,软盘为20个;/MU不装载DELWATCH TSR,除非对它有足够的上端内存使用;/ML即使有上端内存空间,DELWATCH TSR也会被装入常规内存。

C>DISKMAP [/D]drive✓

备份磁盘文件分配表,命令之后可以跟随盘符,/D删除现有的DISKMAP文件。

C>DISKOPT [/B]✓

重新优化磁盘、安排文件,/B指定为单显。

C>DOSBOOK commandfile /B✓

该命令后缺省参数时为激活DR DOS使用帮助信息,也可以在命令之后跟入某一个命令名以获得这个命令的帮助信息,/B为强制菜单以黑白方式显示。

C>EDITOR file✓

这是一个全屏编辑程序,基本兼容了WordStar编辑软件,具体操作方法请参照第五个问题“EDITOR全屏编辑程序”。

C>FC [/A][B][C][L][LB][N][T][W][X] File1 File2✓

比较ASCII文本或二进制文件差别,/A简化输出所有ASCII差异表;/B强使单字符比较;/C忽略大小写差异;/L迫使在ASCII模式下进行;/LB设置最大连续差异行数;/N在ASCII输出中显示行号;/T比较时直接处理制表符而不是用8个空格替代它;/W把所有连续的制表符或空字符压缩为空字符。

C>FILELINK [DIR][DUP][QUI][REC][SET][SLA][TRA][wildspec1][wildspec2][comparm]✓

传输两台计算机之间文件。有关命令缩写:

DIRECTORY(DIR)——DUPLICATE(DUP)——QUIT(QUI)——RECEIVE(REC)——SETUP(SET)——SLAVE(SLA)——TRANSMIT(TRA); wildspec1 原驱动器路径和文件名; wildspec2 目标驱动器路径和文件名; comparm 通信口和波特率,格式

为com#,波特率值。

PC>HILOAD file 把程序装入上端内存。

PC>LOCK 设置口令暂时将操作系统锁定。

PC>MEMMAX[+L][-L][+U][-U][+V][-V]
[+L][+U][+V] 有选择地禁止增强内存功能,防止某些程序发生冲突。

±L 允许/禁止访问低内存;±U 允许/禁止访问高内存;±V 允许/禁止访问由 EMM386.SYS 的/VIDEO 选项保留的内存;/L 显示低内存状态;/U 显示高内存状态;/V 显示 VIDEO 内存状态。

C>MOVE 把文件和子目录转移到别处。

C>PASSWORD[R:password][W:password]
[D:password][P:password][G:password][N]/
NP][NG][S] 对文件和子目录设置口令,防止别人使用你的子目录或拷贝、删除文件。/R 读、删、写、改、拷贝时必须使用口令;/W 只能读和拷贝,删除、改名、改变属性时必须使用口令;/D 建立属性为 D;/P 针对子目录;/G 设置全程缺省口令;/N 删除任何口令;/NP 删除子目录口令;/NG 删除全程缺省口令;/S 在当前目录中文件和子目录进行口令操作。

C>REPLACE 复制选定文件。
C>SCRIPT 提供 PostScript 支持。
C>SETUP 设置系统值。
C>SID 这个程序相当于 DOS 3.3 的 DEBUG 程序,但命令名和使用方法都有所增强,与 DEBUG 区别较大,具体操作命令请参照第六个问题“调试程序 SID”。

C>SSTOR 启动 SuperStor 数据压缩程序。

C>SUPERPCK 启动 SuperPC-Kwik 磁盘缓存程序。

C>TASKMAX 不同程序间进行任务切换。

C>TOUCH 重置文件时间和日期标签。

C>UNDELETE 恢复已删除的文件。

C>UNFORMAT 恢复已格式化的磁盘。

C>UNINSTAL 恢复老操作系统。

C>XDEL 删除子目录和文件组。

C>XDIR 有选择地显示目录文件。

(三)系统配置命令和设备驱动文件

系统配置主要指 CONFIG.SYS 中用到的文件和命令。

1. 系统配置命令

label 为一个标号,可让 GOTO 语句定义进入。

BREAK on/off 可允许用户用 Ctrl+Break 或 Ctrl+C 终止程序。

BUFFERS=nn 开辟一定数量的缓冲区,nn 为 3~99。

CHAIN=路径\文件 在指定路径寻找文件并进行处理。

CLS 清除屏幕。

COUNTRY=XXX[YYY][,Filespec] 指定国家

代码页,XXX 国家电话系统国际代码,YYY 所需代码页,含国家信息文件的路径和文件名。

CPOS nn,mm 将光标定位至 nn 行 mm 列。

DEVICE=Filespec[Switches] 装载设备驱动程序,Filespec 指定设备驱动程序路径和文件名,Switches 设备驱动程序文件相应开关。

DRIVPARM=/C/D:xxx/F:Y/A:zz/1/N/S:ss/
T:tracks

2. 设备驱动文件

ANSI.SYS 屏幕显示、光标和键盘驱动,主要指 \$E[……]的控制。

DEVSWAP.COM 重新分配驱动器号。

DISPLAY.SYS 为 EGA 和 VGA 显示代码页的切换。

DRIVER.SYS 支持类型不同驱动器。

EMM386.SYS 此设备驱动程序可以更好地利用 386 以上系列微机的可用内存。

HIDOS.SYS 用此程序可把 DR DOS 内核重定位至高内存。

PRINTER.SYS 是允许打印机进行代码页功能切换的设备驱动程序。

SSTORDRV.SYS 为支持 Superstor 程序的设备驱动程序。

CDISK.SYS 可以匀出一块内存,开辟快速盘(或虚拟盘)。

五、全屏编辑 EDITOR

EDITOR 是一个纯英文全屏编辑程序,与 EDLIN 程序相比,功能要强得多。

存盘退出:Ctrl+KX 存盘并退出 DOS,Ctrl+KS 存盘后继续编辑,Ctrl+KD 存盘返回 EDITOR 初始状态,Ctrl+KQ 不存盘退出 DOS 状态。

光标控制:Home 或 Ctrl+QR 文件首、End 或 Ctrl+QC 文件末、Ctrl+QS 行首、Ctrl+QD 行末、← 或 Ctrl+S 左移一个字符、→ 或 Ctrl+D 右移一个字符、↑ 或 Ctrl+E 上移一行、↓ 或 Ctrl+X 下移一行、Ctrl+A 左移一句、Ctrl+F 右移一句、PgUp 或 Ctrl+R 上翻一页、PgDn 或 Ctrl+C 下翻一页。

块操作:Ctrl+KB 定义块首、Ctrl+KK 定义块尾、Ctrl+KV 移动块、Ctrl+KY 删除块、Ctrl+KC 复制块、Ctrl+KW 写块到一个文件、Ctrl+KR 读入文件。

删除:Del 或 Ctrl+G 删除光标处字符、←(回删键)或 Ctrl+H 删除光标左边字符、Ctrl+Y 删除一行、Ctrl+T 删除光标右边半行。

其它操作:Ins 或 Ctrl+V 插入或修改开关、Enter 或 Ctrl+N 在光标插入一个新行、Ctrl+J 获得帮助信息、Ctrl+P 光标处输入控制字符。

六、调试程序 SID

SID 命令集:

A 汇编程序。

B 比较两个内存块并在屏幕上显示差异点。

D 显示十六进制内存内容和对应的 ASCII 码。
 E 把文件调入内存。
 F 用字节或字填充到一个内存区域。
 G 从断点执行内存程序。
 H 进行算术运算。
 I 在程序段前缀处准备一个文件控制块和前缀缓冲区,将信息拷贝到最新调入文件的程序段前面。
 M 把内存数据块传送到另一个区域。
 P 清除、设置和显示通过点。
 Q 退出 SID。
 QI 从一个端口读入数据。
 QO 将数据写至一个端口。
 QR 读给定磁盘的绝对地址数据。
 QW 将数据写至给定磁盘的绝对地址。
 R 把文件读入内存的连续块内。

S 修改内存中字节或字内容。
 SR 在给定的内存区域中查找 ASCII 码值,并列出其值地址。
 T 跟踪执行程序。
 U 与 DEBUG 的 U 命令相似,但可显示第一条指令的 CPU 状态。
 V 显示 E 或 R 调入内存文件的首尾地址。
 W 将文件写至首、尾地址。
 X 检查和修改调试过程中的 CPU 状态。
 Z 显示用于算术运算的 8087 协处理器寄存器中内容。

总之,DR DOS 6.0 有着很强的系统管理功能,且它本身就有提供用户操作学习的信息,由于篇幅有限,不便详细介绍,敬请读者谅解。

扩展或扩充内存的使用 ——多任务操作

钟云辉 中国西昌卫星发射中心(615000)

现在,386 微机广泛进入中国的家庭和单位,并且 386DX 微机一般都将四兆内存作为标准配置(386SX 一般为两兆内存)。但是,现在的多数软件都只能运行在 640K 基本内存中,其它内存并没有使用,一部份用户将其扩展或扩充内存化成一个虚拟盘来使用,也算是较“充分”地使用了内存。笔者根据自己对计算机的使用及系统配置体会,从多任务角度谈如何利用 386 的扩展或扩充内存。

大家都知道,386 微机有两种工作模式,即:实地址模式;保护的虚地址模式。在单用户单任务使用时,处于实地址模式,这时直接可访的内存实空间只有 1MB,在这样的情况下,386 微机只不过成了一个速度很快的 8088 PC 机而已。在多用户或多任务使用时,处于保护的虚地址模式,这时实存空间可达数兆,每个任务拥有的空间可达数千兆。理所当然,处于保护的虚地址模式下可以充分利用内存。但是,大多数 386 微机都是作为单机使用,在这种情况下,多数微机并没有使用扩展或扩充内存,笔者推崇在单机上进行多任务操作,可充分使用内存。具体阐述如下:

一、选用具有多任务的高版本 DOS 操作系统

现在 DOS 操作系统广泛使用的有 MS DOS 5.0、MS DOS 6.0、DR DOS 6.0,这些高版本 DOS 操作系统都具有多任务操作功能。本文选用的是 DR DOS 6.0。

二、对计算机系统进行合理配置

对计算机进行合理配置非常重要,因为不管您的微机有多大的内存,但软件一般都需要在 640K 的基本内存中运行,配置好系统的基本出发点在于:尽可能

的使基本内存增大。请看在 DR DOS 下 CONFIG.SYS 文件的配置:

```
hidos=on(将 DR DOS 内核重定位到高位内存)
hidevice=ansi.sys
hidevice=emm386.sys /frame=none /exclude=
a000-b0ff /include=b100-b7ff /exclude=b800-
bfff /rom=c000-c7ff /include=c800-ffff /rom=
f000-f7ff /bdos=ffff
```

(更好地利用 386 的可用内存,地址可修改)

```
hidevice=c:\dos\sstordrv.sys(扩充硬盘容量)
```

```
hidevice=c:\dos\vdisk.sys 1500/e 512,
```

(设置虚拟盘可提高任务切换速度,也可作为它用)

```
files=34
```

```
device=acache.sys 512/p8/e/cw/sw
```

(设置高速缓存区,提高磁盘存取速度)

注:hidevice=xxxx,用于将内存驻留程序或其它实用程序装入上部内存并执行。

DR DOS 如果仅用上述配置中的第一、第三两条,启动微机后,基本内存可有 627K

启动的 AUTOEXEC.BAT 其内容如下:

```
@echo off
```

```
prompt $p $g
```

```
hiloal c:\mouse\mouse
```

(装入鼠标驱动程序)

```
hiloal c:\dos\share
```

(装入文件锁定及共享程序)

```
hiloal c:\dos\taskmax /d=e;
```


(装入多任务操作系统,任务切换在虚盘 E 上进行)

echo on

注:hload xxxx 表示将其程序的驻留部份放在扩展或扩充内存中,少占用基本内存,在编制其批处理时,也可这样使用。

系统启动后,基本内存还有 590K。当然若用户只用 CONFIG. SYS 文件中第一、三条,且取消自动批处理文件中的鼠标驱动程序,内存还可以增加。

三、进入多任务操作

启动上述批处理文件后,系统便进入了多任务环境,DRDOS 可进行多达 20 个任务的切换。本文中将其任务切换在虚拟盘上进行,用户也可将其在硬盘上进行,若用户使用的程序较大,最好在硬盘上进行任务切换,但用户必须修改 TASKMAX. INI 文件的相应部份。

四、多任务操作的好处

多任务操作,实际上就是单机进入保护的虚地址方式下运行的系统之一,最大的好处在于充分利用了 4 兆内存。下面举三个例子说明:

(1)在汉字系统下我们进行了文字处理后,我们要运行 Turbo C 软件,但这时基本内存不够,通常的方

法是退出汉字系统后再进行,若汉字系统能退出还行,若不能退出则要重新启动微机,在多任务下,我们只将汉字系统当作一个任务挂起来,按热键(Ctrl+Esc)再增加一个,在这个任务下运行 Turbo C,这时相当于重新启动了微机,可以检查内存与刚启动后是一样大小。

(2)我们在运行有的软件时,会出现系统死锁,这时不得不重新启动,但在多任务下,我们可以将该任务删出,继续进行其它软件的运行。

(3)我们在调试汇编源程序时,经常会出现程序出错,这时就要回到编辑状态下修改后再进行调试,在多任务下我们将调试环境与程序编辑状态作为两个任务,程序修改后存盘不退出,马上切换到调试环境进行调试,若再出错,马上切换到编辑状态下进行修改后存盘不退出,再切换到调试环境下进行调试,效率非常高。

多任务的优点不仅上述几点,具体方法可参见有关资料。

以上仅从多任务的角度来谈充分利用内存,供用户参考,事实上计算机的内、外存是计算机的重要组成部分,其使用弹性大,使用得好与否,与计算机用户对软硬件知识的掌握有关,欢迎同行共同探讨。

2709 病毒的分析 and 诊治

万武庆 江西抚州地区建行微机室(344000)

计算机病毒发现的历史并不算长,却层出不穷地出现有二千多种,而且每年又有新的病毒产生。笔者最近在机房发现一种新型引导型病毒,用公安部的 SCAN, KILL 和 Carmel 公司的 ANTVIRUS, 包括美国中心点公司检测 1009 种病毒的 CPAV 杀毒软件都不能发现和消除。美国 McAfee Associates 的 Scan 9.11 v100 为了对付不知的病毒,将在 DOS Boot 区发现的未知病毒称为 GenB 病毒(Generic boot),在硬盘 MBR 发现的未知病毒称为 GenP 病毒(Generic MBR)。Scan 9.11 v100 可以从盘中检查出这种未知病毒,但用 SCAN nul/m 命令发现不了内存里有病毒,而且用它提供的 CLEAN 9.11 v100 也不一定能安全地消除病毒。

一、病毒特性

这种引导型病毒感染软盘根区和硬盘主引导区,不感染文件。对于软盘,病毒将真正的引导区移到软盘的 1 头 27 道 9 扇区(故称 2709 病毒,磁介质为 F9 即 5 英寸高密盘移到 1 头 4F 道 0F 扇区),将病毒体写在 1 头 27 道 8 扇区(磁介质为 F9 的在 1 头 4F 道 0E 扇区),并修改根区前几十个字节。对于硬盘,则把主引导区移到隐含的第三扇区,病毒体占据隐含的第二扇区,同时修改主引导区的字节,首部两字节也变为 EB34。

病毒通过染毒的盘启动后进入内存,驻留在 9F80;段地址处(随内存不同而不同),修改了 INT 13 中断向量,若读写盘就感染病毒。

二、病毒破坏性

(1)病毒体和正确的引导区占据磁盘的两个扇区,并覆盖其中的数据,而造成数据和程序的破坏,不可恢复。

(2)对于装有 XENIX 和 DOS 的微机,硬盘染毒则两种操作系统都不能用硬盘启动。

(3)病毒发作破坏文件链,格式化磁盘,属恶性病毒。

三、病毒检测

(1)查内存容量,要比正常少 2K 字节。

(2)检查内存 C>DEBUG 若 9F80;段代码为以下数据则存在此病毒。

-d 9f80,0000

9f80,0000 2E 8C 1E 0E 06 0E 1F A3-06 06 89 1E 08 06 89 0E

9f80,0010 0A 06 89 16 0C 06 8C 06-10 06 89 36 14 06 89 3E

(3)用 Scan 9.11 v100 检查硬盘和软盘,会分别报有 [GenP]和[GenB]病毒。

(4)运行 PCTools,用磁盘功能 edit,会发现软盘(5 英寸低密为 719 扇区,5 英寸高密为 2399,3 英寸高密为

1430)为正确的引导区,而5英寸低密718扇区(5英寸高密2398,3英寸高密1429)为病毒体,同内存中9F80:0000起的一个扇区代码相同。

(5)读入硬盘主引导区或软盘引导区,比较偏移地址03F起到067间字节,代码被改为

```
1137:0030 FA 33-C0 8E D0 BC 00 7C 16 1F
1137:0040 FB A1 13 04 48 48 A3 13-04 B1 06 D3 E0 8E C0 A3
.....
```

C)DEBUG

-L 100 0 0 1 ;查看A驱根区013F~0167间代码
-a 300

```
1137:0300 mov ax,0201
1137:0303 mov bx,0500
1137:0306 mov cx,0001
1137:0309 mov dx,0080
1137:030C int 13 ;读硬盘主引导区
1137:030E int 3
1137:030F <cr>
```

-g=300

-d 500 ;查看053f~0567间代码,已更改

四、病毒消除

(1)用McAfee Associates的Clean 9.11 v100试着杀毒。

在干净系统下 clean c:[genp]

clean a:[genb]

但由于Clean对这未知病毒无针对解法,故由于某些原因(如软盘是用PC Tools格式化的),Clean常报“Virus cannot be safely removed from boot sector”则用手工解毒。

(2)弄清了病毒原理,解毒就容易了,只要用干净的系统盘启动后,用DEBUG把正确的引导区或主引导区写回0面0道1扇区就行。步骤如下:

对于软盘(把软盘插入A驱) 对于硬盘

C)debug

C)debug

```
a 100
0100 mov ax,0201
0103 mov bx,0500
0106 mov cx,2709
;若F9介质,改为 mov cx,4f0f 0109 MOV DX,0080
0109 mov dx,0100 010C int 13
;若B驱,109句改为 mov dx,0101 010E int 3
010C int 13 010F <cr>
010E int 3 -g=100 ;读出正确MBR
010F <cr> -e 102 03
-g=100 <cr> -e 107 01
-W 500 0 0 1 <cr>,更正引导区
;若在B驱改W500 101<cr>
```

;此时病毒已解,以下步骤可不作

;再把盘内病毒体用F6覆盖 ;再用00覆盖病毒体

```
a 100 -F 500 800 00
```

```
0100 mov ax,0301 -E 107 02
```

```
0103 mov bx,500 -g=100
```

```
0106 mov cx,2708
```

```
;F9改 mov cx,4foe -Q
```

```
0109 mov dx,0100
```

```
;B驱改 mov dx,0101
```

```
010C int 13
```

```
010E int 3
```

```
010F <cr>
```

```
f 500 800 f6
```

```
g=100
```

Q;退出已清毒

五、病毒预防

在内存干净和主引导区、引导区正确的时候,运行CPAV中的BOOTSAFE.EXE自动建立备份主引导区和引导区的CBOOT.CPS文件,在AUTOEXEC.BAT中加入BOOTSAFE命令,用硬盘启动,常运行BOOTSAFE命令,若主引导区有改变,运行BOOTSAFE/R恢复,保护硬盘MBR。勿用外来、未封口的软盘启动,若启动时A驱误插软盘,则应重新启动,防止病毒已入内存。

中华学习机做简易字幕机

李建林 陕西省人民警察学校电教室(710054)

现在,不少中学都有中华学习机和录像机,用于计算机教学和播放录像。但是,一般学校中不会有字幕机的。中华学习机内固化有国标一、二级汉字,使用汉字很方便。把中华学习机做为一台简易字幕机,在中文状态下把需要显示的文字在屏幕上显示出来,配合播放录像使用,将收到很好的效果。

为了使用方便,使汉字显示效果更好,笔者将放大汉字的机器码程序输入计算机中,调试好后存入磁盘,建立一张设计字幕专用磁盘。用它可以设计出比较满

意的字幕来。下面将笔者具体应用情况介绍出来,有兴趣有条件者不妨一试。

一、建立一张设计字幕的专用磁盘

操作步骤如下:

1. 格式化一张新盘

用系统盘装入DOS后,键入HELLO程序(见程序一),格式化一张新盘。以后用该盘开机引导时,执行HELLO程序后,计算机就自动进入中文状态、将放大汉字的机器码程序读入内存、完成三个内存单元所必

需的数值设置、并列出磁盘目录,以供选择。

2. 键入放大汉字的机器码程序

在监控状态下,将放大汉字的机器码程序(见程序二)输入计算机,检查调试好后,以 FDHZ 为文件名存入磁盘,具体操作是 BSAVE FDHZ A \$ 1B63,L \$ 480 这样,就建立了一张供设计字幕用的专用磁盘。

二、设计字幕

当需要设计字幕时,把专用磁盘插入驱动器内,开机引导后,计算机自动进入中文状态,放大汉字的机器码程序就自动读入内存。这时,用 BASIC 语言和中华学习机的 & 命令就可设计字幕了。具体 & 命令如下:

& W 0,255,0,255 设定汉字显示的窗口座标
& D X,Y 将汉字显示在 X,Y 座标处
& C F 使屏幕显示黑底白字
& S X,Y,Z,H,C 将汉字横向放大 X 倍,纵向放大 Y 倍,字间距为 Z,行间距为 H,颜色为 C(其值为 0)

& P "....." 显示引号间的汉字

& P A \$ 显示字符串 A \$ 的内容

下面提供一个实际应用例子供参考(见程序三)。

三、字幕使用

1. 直接由计算机显示输出

用射频电缆线把中华学习机射频输出端和录像机的射频输出端,连接在一个带开关三通的两个输入端上,三通的输出端连接在电视机上。当要使用字幕时,控制三通上的开关,使计算机与电视机接通,运行字幕程序,显示输出字幕。当字幕显示完后,要播放录像时,只要拨动开关,接通录像机就可以了。

2. 把字幕录在磁带上,通过放录像显示

用视频电缆线把中华学习机视频输出端,连接在录像机的视频输入端,运行字幕程序,显示出字幕,按下录像键,即可将字幕录在磁带上,以备播放时用。字幕持续时间自己灵活掌握。用这种办法,可以给自制的录像资料加上字幕。

程序一 HELLO

```
10 D$=CHR$(4)
20 PRINT D$ "PR#3"
30 PRINT D$ "BLOAD FDHZ"
40 POKE 1013,76
50 POKE 1014,99
60 POKE 1015,27
70 PRINT D$ "CATALOG"
80 END
```

程序三

```
10 REM QWXY
20 HGR2
30 &W0,255,0,255
40 &CF
50 READ A$,B$,C$,D$,E$
```

```
60 &S2,2,1,2,0:&D0,0
70 &PA$
80 &S2,2,1,2,0:&D0,40
90 &PB$
100 &S2,2,1,2,0:&D0,80
110 &PC$
120 &S2,2,1,2,0:&D0,120
130 &PD$
140 &S1,1,4,0,0:&D96,176
150 &PE$
160 POKE-16368,0
170 K=PEEK(-16384)
180 IF K<>155 THEN 160
190 DATA "请勿吸烟","请不要随地吐痰"
200 DATA "请保持室内清洁","请自觉遵守"
210 DATA "———电教室"
220 END
```

程序二 FDHZ

1B63-	48	20	B1	00	68			
1B68-	C9	44	D0	03	40	A4	1B	C9
1B70-	53	D0	03	4C	E0	1B	C9	50
1B78-	D0	03	4C	1F	1E	C9	57	D0
1B80-	03	4C	B4	1B	C9	47	D0	03
1B88-	4C	09	1C	C9	43	D0	03	4C
1B90-	16	1C	A2	10	4C	12	D4	20
1B98-	67	DD	20	08	E1	A2	00	A5
1BA0-	A1	A4	A0	60	20	97	1B	8C
1BA8-	C8	1F	8D	C4	1F	20	4C	E7
1BB0-	8E	C5	1F	60	20	97	1B	8D
1BB8-	C6	1F	8D	C4	1F	8C	C7	1F
1BC0-	8C	C8	1F	A9	2C	20	C0	DE
1BC8-	20	97	1B	20	6E	1D	A5	08
1BD0-	8D	B3	1D	20	4C	E7	8E	C5
1BD8-	1F	20	4C	E7	8E	B7	1D	60
1BE0-	20	F8	E6	86	05	20	4C	E7
1BE8-	86	06	A9	10	20	31	1C	8D
1BF0-	B5	1D	20	4C	E7	8E	B8	1D
1BF8-	20	4C	E7	8A	18	6D	B5	1D
1C00-	8D	B6	1D	20	4C	E7	86	E3
1C08-	60	20	F8	E6	8A	0A	0A	0A
1C10-	0A	0A	85	E6	60	60	20	B7
1C18-	00	C9	46	D0	07	A9	00	8D
1C20-	D1	1F	F0	09	C9	4F	D0	08
1C28-	A9	FF	8D	D1	1F	4C	B1	00
1C30-	60	85	00	CA	F0	05	18	65
1C38-	00	90	F8	60	18	AD	C5	1F
1C40-	6D	B6	1D	8D	C5	1F	AD	C6
1C48-	1F	8D	C4	1F	AD	C7	1F	8D
1C50-	C8	1F	60	A2	35	4C	12	D4
1C58-	A9	00	85	07	20	3A	1D	A5
1C60-	1C	CD	B7	1D	B0	ED	18	AD
1C68-	C4	1F	6D	B2	1D	85	1B	85
1C70-	00	90	03	EE	C8	1F	20	63
1C78-	1D	A5	08	CD	B3	1D	90	06

1C80-	20	3C	1C	18	90	D6	AD	C4
1C88-	1F	85	00	20	63	1D	20	15
1C90-	1D	20	C5	1D	A5	05	85	02
1C98-	AD	C5	1F	85	01	A2	08	AD
1CA0-	B4	1D	85	1F	A5	06	85	03
1CA8-	06	1F	08	86	04	A6	01	20
1CB0-	89	1D	A6	04	A4	08	A5	09
1CB8-	28	08	90	0F	A5	09	25	FC
1CC0-	F0	07	11	26	91	26	18	90
1CC8-	08	A5	09	49	FF	31	26	91
1CD0-	26	E6	01	C6	03	D0	D4	68
1CD8-	CA	D0	C9	E6	00	06	09	10
1CE0-	0C	A9	01	85	09	E6	08	20
1CE8-	4E	1D	20	15	1D	C6	02	D0
1CF0-	A7	E6	07	A5	00	C5	1B	D0
1CF8-	92	A9	10	85	07	A5	01	8D
1D00-	C5	1F	C5	1C	90	80	A5	1B
1D08-	8D	C4	1F	38	A5	1C	ED	B5
1D10-	1D	8D	C5	1F	60	A5	08	85
1D18-	11	A5	E3	0A	46	11	69	00
1D20-	A8	B9	B9	1D	85	FC	60	A4
1D28-	08	A5	FC	10	06	B1	26	09
1D30-	80	D0	04	B1	26	29	7F	91
1D38-	26	60	AD	C5	1F	18	6D	B5
1D40-	1D	85	1C	AD	C4	1F	85	00
1D48-	20	63	1D	20	15	1D	A9	7F
1D50-	C5	FC	F0	0E	AE	C5	1F	20
1D58-	89	1D	20	27	1D	E8	E4	1C
1D60-	D0	F5	60	AD	C8	1F	85	FD
1D68-	A5	00	A2	00	A4	FD	F0	09
1D70-	C9	50	B0	05	A2	23	69	04
1D78-	E8	38	E9	07	B0	FA	96	08
1D80-	69	07	AA	BD	AB	1D	85	09
1D88-	60	8A	29	C0	85	26	4A	4A
1D90-	05	26	85	26	8A	85	27	0A
1D98-	0A	0A	26	27	0A	26	27	0A
1DA0-	66	26	A5	27	29	1F	05	E6
1DA8-	85	27	60	01	02	04	08	10
1DB0-	20	40	10	23	29	10	10	C0
1DB8-	00	FF	FF	55	2A	D5	AA	2A
1DC0-	55	AA	D5	7F	7F	A5	07	4A
1DC8-	4A	4A	4A	B0	04	A0	00	F0
1DD0-	02	A0	01	4A	B0	04	A9	E0
1DD8-	D0	0B	AD	C9	1F	F0	04	A9
1DE0-	F0	D0	02	A9	E9	8D	EC	1D
1DE8-	8D	F0	1D	B9	E0	1F	4A	99
1DF0-	E0	1F	2E	B4	1D	C8	AD	C9
1DF8-	1F	F0	01	C8	CC	D3	1F	90
1E00-	EA	60	A9	00	AA	A8	8D	03
1E08-	C0	BD	00	40	8D	02	C0	99
1E10-	E0	1F	AD	C9	1F	D0	01	E8
1E18-	E8	C8	E0	20	D0	E8	60	A9
1E20-	22	A0	00	8D	CA	1F	D1	B8
1E28-	F0	23	20	E3	DF	84	FB	85
1E30-	FA	A5	B8	8D	CB	1F	A5	B9

1E38-	8D	CC	1F	A0	01	B1	FA	85
1E40-	B8	C8	B1	FA	85	B9	A9	00
1E48-	8D	CA	1F	F0	03	20	B1	00
1E50-	A0	00	B1	B8	C9	22	D0	13
1E58-	20	B1	00	AD	CA	1F	D0	0A
1E60-	AD	CC	1F	85	B9	AD	CB	1F
1E68-	85	B8	60	C9	27	D0	10	AD
1E70-	CD	1F	49	FF	8D	CD	1F	A9
1E78-	01	20	97	1F	4C	50	1E	C9
1E80-	60	D0	0B	20	3C	1C	A9	01
1E88-	20	97	1F	4C	50	1E	C9	7F
1E90-	F0	16	8D	CE	1F	A9	00	8D
1E98-	CF	1F	A9	01	20	97	1F	A9
1EA0-	00	8D	C9	1F	A9	08	D0	0A
1EA8-	20	85	1F	A9	01	8D	C9	1F
1EB0-	A9	10	8D	D3	1F	CE	CE	1F
1EB8-	AE	CF	1F	D0	07	A9	1F	A0
1EC0-	D4	4C	D2	1E	E0	10	B0	06
1EC8-	A9	1F	A0	D8	90	04	A9	1F
1ED0-	A0	DC	84	FA	85	FB	A0	00
1ED8-	B1	FA	8D	F2	1E	C8	B1	FA
1EE0-	8D	11	1F	C8	B1	FA	8D	31
1EE8-	1F	C8	B1	FA	8D	29	1F	38
1EF0-	8A	E9	00	8D	CF	1F	A9	5E
1EF8-	8D	D0	1F	20	AC	1F	18	AD
1F00-	D0	1F	6D	CE	1F	8D	D0	1F
1F08-	A9	00	6D	CF	1F	A8	4A	18
1F10-	69	00	AA	98	29	01	8D	CF
1F18-	1F	A0	05	0E	D0	1F	2E	CF
1F20-	1F	88	D0	F7	18	AD	D0	1F
1F28-	69	00	8D	D0	1F	AD	CF	1F
1F30-	69	00	C9	80	90	03	E9	40
1F38-	E8	8D	CF	1F	AD	D1	1F	29
1F40-	10	8E	D2	1F	0D	D2	1F	8D
1F48-	B0	C0	AD	D0	1F	8D	0A	1E
1F50-	AD	CF	1F	8D	0B	1E	20	02
1F58-	1E	AD	D3	1F	A6	05	20	31
1F60-	1C	8D	B2	1D	A2	90	AD	CD
1F68-	1F	F0	02	A2	B0	8E	BA	1C
1F70-	20	58	1C	AD	C4	1F	18	6D
1F78-	B8	1D	8D	C4	1F	90	03	EE
1F80-	C8	1F	4C	50	1E	A0	01	20
1F88-	A1	1F	8D	CF	1F	A0	02	20
1F90-	A1	1F	8D	CE	1F	A9	03	18
1F98-	65	B8	85	B8	90	02	E6	B9
1FA0-	60	B1	B8	20	92	C3	29	7F
1FA8-	38	E9	20	60	A9	00	A0	08
1FB0-	4E	D0	1F	90	04	18	6D	CF
1FB8-	1F	4A	6E	D0	1F	88	D0	F3
1FC0-	8D	CF	1F	60	00	00	00	00
1FC8-	00	49	D0	31	C1	00	30	38
1FD0-	3A	FF	50	32	00	00	40	20
1FD8-	01	00	60	00	10	02	49	00
1FD0-	52	00	00	FF				

ProDOS 系统内部结构剖析

廖 凯 北京铁路局卫生防疫站(100038)

GOSYSTEM 用于制作所有 MLI 调用,因为错误必须在返回到调用子程序之前被翻译。在进入时,累加器应该包含调用号。根据调用号查寻参数表地址并设置参数表。只有文件管理调用可以利用此子程序: \$C0—\$D3。此 BASIC 系统程序的最初实现只包含这些调用。

```
BE70:GOSYSTEM   STA SYSCALL      ;保存调用号
BE73:           STX CALLX        ;保存 X 寄存器
BE76:           AND # $1F        ;清除调用号的高位
BE78:           TAX              ;并用于查表索引
BE79:           LDA SYSCTBL,X     ;取得参数的低地址

BE7C:           STA SYSPARM
BE7F:           LDX CALLX        ;调用前恢复 X 寄存器

BE82:           JSR MLIENTRY      ;调用 ProDOS MLI 以执行请求

BE85:SYSCALL     DFB 0
BE86:SYSRARM     DW 1            ;(高地址应与参数表相同)
BE88:           BCS BADCALL      ;若遇到错误则转 BADCALL

BE8A:           RTS
```

BADCALL 将 MLI 错误转换为对应的 BASIC 系统程序错误。进入时应将错误号输入累加器。返回时累加器内为 BASIC 系统程序错误号。所有不可识别的错误被标为 I/O 错误。X 寄存器被恢复为它调用前的值。进位标志被置位。

```
BE8B:BADCALL    LDX # $12        ;19 个错误为
BE8D:MLIERR1    CMP MLIERTBL,X   ;非 I/O 错误
BE90:           BEQ MLIERR2
BE92:           DEX
BE93:           BPL MLIERR1
BE95:           LDX # $13        ;若不可识别则成为 I/O 错误

BE97:MLIERR2 LDA BIERRTBL,X; 错误代码返回累加器

BE9A:           LDX CALLX        ;恢复 X 寄存器
BE9D:           SEC              ;设置进位以指出错误

BE9E:XRETURN    RTS
BE9F:BISPARE1   DFB $00
```

下面的是系统调用参数表,这些表必须存在内存的同一页内。这里只标出那些会被修改的参数。SYSCTBL 下面包含每个参数表的低地址。SYSCTBL 被 GOSYSTEM 使用,以建立每个调用的参数表的

地址。

```
BEA0:SCREATE    DFB $07
BEA1:           DW TXBUF-1      ;指针指向路径名
BEA3:CRACCESS    DFB $C3        ;若目录建立则为 $C1
BEA4:CRFILID     DFB $00
BEA5:CRAUXID     DW $0000
BEA7:CRFKINO     DFB 0
BEA8:           DW 0            ;没有预定的日期/时间
BEAA:           DW 0
BEAC:SSGPRFX     EQU *
BEAC:SDSTROY     DFB $01
BEAD:           DW TXBUF-1      ;此调用不要求修改
BEAF:SRENAME     DFB $02
BEB0:           DW TXBUFS-1     ;无修改需要
BEB2:           DW TXBUF2
BEB4:SSGINFO     DFB $00        ;若为 SET-FILE-INFO
                                ;则 P.CNT=7;
                                ;若为 GET-FILE-INFO
                                ;则 P.CNT=A

BEB5:           DW TXBUF-1
BEB7:FIACCESS    DFB $00        ;被上锁/解锁使用
BEB8:FIFILID     DFB $00        ;FILE ID 是类型标识
BEB9:FIAUXID     DW $0000       ;辅助标识记录装入地址和记录长度

BEBB:FIFKIND     DFB $00        ;识别树型目录
BEBC:FIBLOKS     DW $0000       ;被 CAT 命令使用,用于根目录

BEBE:FIMDATE     DW $0000       ;修改日期和时间
BEC0:           DW $0000        ;在调用前应被置为零
BEC3:           DW $0000        ;忽略建立日期和时间
BEC4:           DW $0000
BEC6:SONLINE     EQU *
BEC6:SSETMRK     EQU *
BEC6:SGETMRK     EQU *
BEC6:SSETEOF     EQU *
BEC6:SGETEOF     EQU *
BEC6:SSETBUF     EQU *
BEC6:SGETBUF     EQU *
BEC6:           DFB $02        ;参数个数
BEC7:SBUFREF     EQU *
BEC7:SREFNUM     EQU *
BEC7:SUNITNUM    EQU *
BEC7:           DFB 0            ;设备号或参考号
BEC8:SDATPTP     EQU *
BEC8:SMARK       EQU *
BEC8:SEOF        EQU *
```

```

BEC8;SBUFADR EQU *
BEC8: DFB 0,0,0 ;某些调用只用 2 字节
;MRK 和 EOF 用 3 字节

BEC8;SOPEN DFB $ 03
BECC: DW
TXBUF-1
BECE;OSYSBUF DW $ 0000
BED0;OPEFNUM DFB 0
BED1;231 SNEWLIN DFB $ 03
BED2;232 NEWLREF DFB $ 00 ;参考号
BED3;233 NLINENBL DFB $ 7F ;换行字符总是 CR
BED4;234 DFB $ 0D ;$ 0D 和 $ 8D 都有效
BED5;5 236 SREAD EQU *
BED5;5 237 SWRITE EQU *
BED5;238 DFB $ 04
BED6;239 RWREFNUM DFB $ 00
BED7;240 RWDATA DW $ 0000 ;指向被读写的数据
BED9;241 RWCOUNT DW $ 0000 ;被读写的字节数
BEDB;242 RWTRANS DW $ 0000 ;送回读写的字节数
BEDD;D 244 SCLOSE EQU *
BEDD;D 245 SFLUSH EQU *
BEDD;246 DFB $ 01
BEDE;247 CFREFNUM DFB $ 00
BEDF;249 CCCSPARE DFB $ 00
BEE0: ASC "COPYRIGHT APPLE, 1983"
BEF5;GETBUFR JMP GETPAGES
BEF8;FREBUFR JMP FREPAGES
BEF8;RSHIMEM DFB 0,0,0,0,0

```

1. 调用 BASIC 系统程序

有时程序要执行已被 BASIC 系统程序实现的功能——删除和改名文件、显示一个目录等等。DOSCMD 向量正提供此功能。

首先程序应该放置所需的 BASIC 命令到输入缓冲区内(\$200)。它应是一个 ASCII 字符串,并且高位元被设置,后跟一个回车字符(\$8D),完全与监控程序 GENLN 子程序存入的字符串一样。下一步程序应 JSR 到 DOSCMD 进入点(\$BE03)。

BASIC 系统程序将分析此命令,建立所有参数,然后执行此命令。若有错误,它将错误码放入 ER-RCODE 单元(\$BE0F)。若它为 0,则无错误。

要想显示相应的错误信息,就需要程序将错误码放入累加器内并执行 PRINTERR 子程序(\$BE0C)。

2. 增加命令到 BASIC 系统程序

在整体页面内的 EXTRNCMD 单元允许你增加命令到 ProDOS 命令集中。

当你输入一个命令时,BASIC 系统程序查看它是否是 ProDOS 命令。若不是,ProDOS JSR 到 EXTRNCMD 单元(\$BE06)以便让程序索取指令。EXTRNCMD 单元应该包含一条 JMP 指令,指向外部命令子程序的起始地址。若没有安装外部命令子程序,EXTRNCMD 则包含一个 JMP 向有一个 RTS 指令的 XRETURN(\$BE9E),将控制权返回给 BASIC 系统程序。

当程序被调用时,未被分析的指令串的首八个字节可以在 VPATH1(\$BE6C)指向的缓冲区内找到。子程序有两个选择:它可以自行分析命令,或可以设置 PBITS 单元并让 BASIC 系统程序分析此命令串。让 BASIC 系统程序分析命令串具有局限性,就是只能使用被其它命令使用的参数(如#S,#D,T 类型等等)。

3. 命令字符串分析

首先,外部命令必须通知 BASIC 系统程序哪些参数为命令所使用。它通过把适当的值分配给两个 PBITS 字节(\$BE54 和 \$BE55,高八位在 \$BE54)来做到,它有以下含义:

位元的含义:

- \$BE54 15:需要取出部首。路径名是任选的。
- 14:无参数被处理。
- 13:命令只有在程序执行期间有效。
- 12:文件名是任选的。
- 11:若文件不存在,则允许建立。
- 10:文件类型(T 类型)任选。
- 9:要求第二个文件名。
- 8:要求第一个文件名。
- \$BE55 7:提供地址(A#)。
- 6:提供字节(B#)。
- 5:提供终止地址(E#)。
- 4:提供长度(L#)。
- 3:提供行号(@#)。
- 2:提供槽口及驱动器号(S#和D#)。
- 1:提供字段(F#)。
- 0:提供记录(R#)。

其次,程序应该将识别的命令字的长度减一放入 XLEN(\$BE52);它应放置一个 \$00 到 XNUM(\$BE53),表示找到一个外部命令,它应将进一步处理已分析的命令的子程序地址存入 XTRNADDR(\$BE50),而后它应 RTS 回到 BASIC 系统程序。

BASIC 系统程序将查看被识别的命令,并且它将按照 PBITS 分析串。对于被命令使用的每个参数,它将设置 FBITS(\$BE56)内相应的位元并修改整体页面内该参数值。最终,它将执行一个 JSR 到 XTRNADDR(\$BE50)内指示的单元。

程序现在可以处理指令。除文件名外,所有参数被存在整体页面内,文件名被存在经 VPATH1 和 VPATH2 指出的单元内。

HELP 命令就是这样的。当你打入一 HELP 时,HELP 命令被装入内存 \$2000 的位置;它将 HIMEM 下移并放置它自身到 HIMEM 上部,然后它在位图内标记它自身。最后放置程序的起始地址到 EXTRNCMD 向量。BASIC 系统程序现在可识别一系列 HELP 命令和 NOHELP 命令,NOHELP 命令从 EXTRNCMD 向量删除 HELP 程序的地址,从位图删除程序标记,并将 HIMEM 上移。

下面的例子说明如何使用一个外部指令,这个命令叫作 BEEPSLOT,是一个无意义的命令;它确认命

令字(BEEPSLOT),查看缺省槽口号是什么,然后它按槽口号发出一段声音。这程序说明在一个外部命令内部怎样使用整体参数(V SLOT)。

```

0300:      0300 1      ORG $300
0300:      2      MSB ON
0300:      FF3A 3 BELL EQU $FF3A;监控
                                BELL 子程序
0300:      FCA8 4 WAIT EQU $FCA8;监控
                                WAIT 子程序
0300:      BE61 5 V SLOT EQU $BE61;CI 槽口
0300:      0200 6 INBUF EQU $200;系统输入缓冲区
0300:      7 *
0300:A2 00      8      LDX #00;字符偏移
0302:BD 00 02  9 NXTCHR LDA INBUF,X;取第一字符
0305:DD 25 03 10      CMP CMD,X;它符合码?
0308:D0 19      11      BNE TRYNEXT;不符,试下一个
030A:E8         12      INX      ;下一个字符
030B:E0 08      13      CPX # $08;读了 8 个字符码?
030D:D0 F3      14      BNE NXTCHR;不是,读下一个
030F:AD 61 BE 15 GOODCMD LDA V SLOT;取槽口号
0312:29 0F      16      AND # $0F;置高位为零
0314:AA         17      TAX      ;作为索引保存

```

0315:F0 0D 18 NXTBEEP BEQ DONE;索引值=0,结束发声

```

0317:20 3A FF 19      JSR BELL;否则,发声一次
031A:A9 B0      20      LDA # $B0;取得延迟
031C:20 A8 FC 21      JSR WAIT;并等待
031F:CA         22      DEX      ;索引值减 1
0320:4C 16 03 23      JMP NXTBEEP;并重复
0323:60         24 TRYNEXT RTS ;无下一个命令
0324:60         25 DONE RTS ;全部结束
0325:          26 *
0325:C2 C5 C5 D0 27 CMD ASC "BEEPSLOT"

```

若没有其它的外部命令程序安装,则只修改指示程序的起始地址的 EXTRNCMD 单元。若有其它的命令安装并且命令字不符,你必须修改这程序以转到下一个程序的开端。

例如,若 HELP 程序被安装,则它在单元 HELP 开始。若命令字不符就要修改 BEEPSLOT 以便它调用 HELP,修改行号 23 为:

23 TRYNEXT JMP HELP;再试 HELP 命令
如此外部命令可以被一起串连起来。

DOS 操作系统分析讲座

第二讲 键盘输入系统(上)

崔来堂 河北石家庄铁道学院计算机教研室(050043)

键盘(Keyboard)是微机系统中常用的标准输入设备,用户通过键盘向微机输入程序和数据,发布操作命令,指挥微机运行。因此,键盘是一种重要的人机介面。

一、键盘输入系统的硬件组成

由于键盘操作速度慢,随机性强,为了保证 CPU 的工作效率,主机对键盘采用中断管理方式。键盘输入系统的硬件,由键盘、键盘接口、中断控制和数据通路等部分组成,各部分之间的关系示于图 2—1。

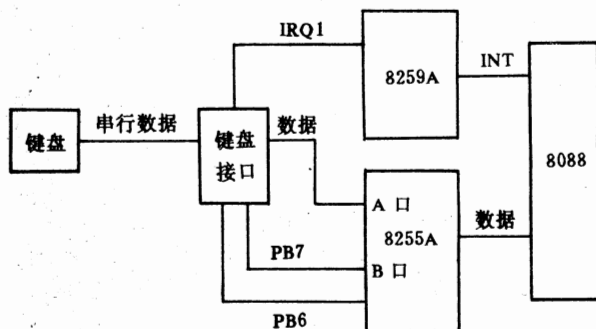


图 2—1 键盘输入系统的硬件组成

键盘是一个独立装置,它的工作由装在它面里的 MCS48 或 51 系列单片机进行管理。键盘与主机之间通过一条 5 芯电缆进行连接,该电缆实用 4 根线:时钟信号线、数据线、地线和电源线正极。

操作键盘时,它里面的单片机通过电缆,向主机中的键盘接口串行发送 8 位的“键盘扫描码”;键盘接口把串行码转换为并行码,每转换完一个 8 位键盘扫描码,便通过中断控制芯片 8259A 向 CPU 发出一个中断请求 IRQ1;CPU 响应后,立即进入相应的中断处理程序,从 8255A 的 A 口取走并行的键盘扫描码,并通过 8255A 的 PB7 清除键盘接口,以便准备接收下一个键盘扫描码,图中 8255A 的 PB6 线,是与 PB7 配合,在 CPU 对键盘初始化后,给出“允许键盘工作”状态。

二、键盘缓冲区及其工作原理

由于键盘操作与主机不同步等原因,必须在内存设置缓冲区,与键盘有关的缓冲区共分为三级:一级键盘缓冲区,是为 ROM—BIOS 的键盘中断程序而设置的,直接与硬件相关,长 32 字节;再者,为了使用户能方便地对键入的内容进行修改,DOS 又在自身设置了二级键盘缓冲区,长 128 字节;此外,有的键盘功能调用,要求用户在应用程序中再设置缓冲区,是为三级键盘缓冲区,长度按实际情况设定,三者的相互关系是:键入的内容首先送到一级缓冲区,DOS 自动将对应的

字符码送入二级缓冲区,编辑修改后打回车认可,必要的话,通过有关中断调用,DOS 再将二级缓冲区的字符串,连同回车一起,送入三级缓冲区,提供给应用程序使用。

因为与键盘操作直接有关的是一级缓冲区,限于篇幅关系,下面仅对这个缓冲区进行较为详细的分析讨论。

在 RAM 内存的 40H 段,是一个数据区,专门用于记录一些参数或进行通信,称为“DOS 通信区”,其中,有一个 32 字节的区域(40:1EH—40:3DH),就是上述的一级键盘缓冲区,为方便计,以下简称其为“键盘缓冲区(Keyboard Buffer)”,它直接接收键盘键入的信息,该缓冲区被设计成先进先出的循环队列,由四个指针控制它的工作,其结构请见图 2—2。

低字节 高字节

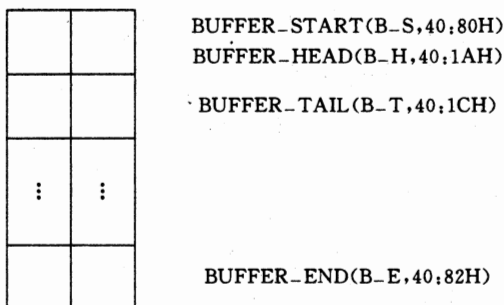


图 2—2 键盘缓冲区(循环队列)

指针 BUFFER_START 和 BUFFER_END(简记为 B_S 和 B_E)在内存的 40:80H 和 40:82H 两个字中,它们是静态指针,分别指向缓冲区的头和尾; BUFFER_HEAD 和 BUFFER_TAIL(简记为 B_H 和 B_T)在内存的 40:1AH 和 40:1CH 两个字中,它们是动态指针,分别指向缓冲区中待取和待存信息的位置,键盘缓冲区的操作过程示于图 2—3。

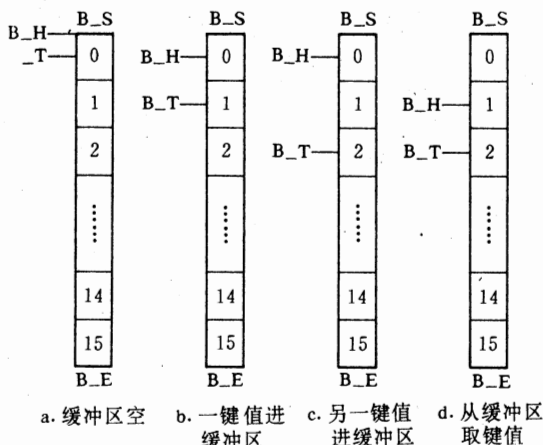


图 2—3 键盘缓冲区操作示意图

初始, B_H 与 B_T 重合, 缓冲区为字, 按动按键后, 扫描码送入 8255A 的 A 口, DOS 立即调用硬件中断 INT 9H, 取出扫描码, 转换为字符码(即相应的 ASCII 码), 字符码存入指针 B_T 所指字的低字节, 扫描码送入高字节; B_T 指针值增 2。当 B_T 值逐步增至缓冲区末端, 即等于 B_E 时, 将自动绕到缓冲区始端, B_T 等于 B_S; B_T 继续转移, 当它等于 B_H 时, 表示缓冲区已满。

需要读取键盘缓冲区的内容时, DOS 调用软中断 INT16H 的 0 号子功能, 取出指针 B_H 所指字的内容, 低字节的字符码送 AL, 高字节的扫描码送 AH, 并使 B_H 的值增 2; 当 B_H 增至与 B_T 相等时, 表示缓冲区内容已被取空; 如果 B_H 移到了缓冲区末端, 也将自动绕至始端。

下面几段程序用于描述键盘缓冲区循环队列操作:

```

; 初始清缓冲区
MOV BX, BUFFER_START
MOV BUFFER_HEAD, BX      ; 使 B_H = B_S
MOV BUFFER_TAIL, BX      ; 使 B_T = B_S
; AX 中的字符码送缓冲区
MOV BX, BUFFER_TAIL      ; 取尾指针
MOV SI, BX               ; 暂存
CALL K4                  ; 尾指针准备移动
CMP BX, BUFFER_HEAD      ; 与头指针相等吗?
JE K6                    ; 等, 缓冲区满
MOV [SI], AX             ; 不等, 送字符码
MOV BUFFER_TAIL, BX      ; 尾指针移动
; 从缓冲区中取字符码送 AX
K1: MOV BX, BUFFER_HEAD  ; 取头指针
CMP BX, BUFFER_TAIL      ; 与尾指针相等吗?
JZ K1                    ; 等, 区空, 循环
MOV AX, [BX]             ; 不等, 取字符码
CALL K4                  ; 头指针准备移动
MOV BUFFER_HEAD, BX      ; 头指针移动
; 指针增量
K4 PROC NEAR
INC BX
INC BX                    ; 指针值增 2
CMP BX, BUFFER_END      ; 到缓冲区末端吗?
JNE K5                  ; 不到, 继续
MOV BX, BUFFER_START    ; 到, 绕至始端
K5: RET
K4 ENDP
; 缓冲区满处理
K6: .....

```

对下述几个问题进一步分析说明:

(1) 中断 INT 9H 对扫描码分析时, 首先检查换挡键或双态键是否设置, 以获取正确的字符码(例如, 是 'a' 还是 'A'?)。

(2) 当按下的是功能键、光标键或某些组合键时, INT 9H 对扫描码分析后, 不能生成标准(下转 11 页)

普及型逻辑笔的制作

车金相 北京宣武区科技青少年馆(100053)

普及型逻辑笔用红、黄、绿三种颜色的发光二极管显示高电平、低电平、高阻及脉冲的有无四种状态,它是学习 TTL 集成电路的好帮手。买一个正规的逻辑笔要花近百元,自己动手,十几元就可以装一个,既经济,又实惠。

电路及原理:电路如图 1 所示,其中 A₁、A₂、A₃ 是四电压比较器 LM339 中的三个。原理请参看 93 年第 9 期《电子与电脑》杂志。

外观与结构:组装好的普及型逻辑笔外观及内部结构如图 2。

元件选择:逻辑笔体积比较小,所以元器件选择时尽量考虑少占空间,电阻选用 1/16 瓦的小电阻,发光二极管的直径采用 3 毫米,二极管选用 IN4148,详见元器件清单。

制作注意事项:

元器件的位置如图 3 所示,图 4 为布线图,印刷线路板上有一个大焊盘,是探针焊接处。

检查印刷线路板:检查从两方面入手,一方面检查有无短路的地方,另一方面检查金属化孔是否正反面全都导通。

熟悉印刷线路板:板上有丝漏网印的面叫正面,即插元件面,插元件可以对照图 3 进行,插 LM339 时一定要小心,集成电路的缺口和板上画的一定要相符。没

印元器件编号的为反面,又称它为焊接面,焊点都在该面上。

焊接顺序:先焊探针,注意让焊好的探针和印刷线路板在同一个水平面内。再焊电阻、电容和二极管,焊二极管 IN4148 时,应先用万用表测一下正、负极,避免接反,焊发光二极管除注意正、负极不要接反外,还要选择适当的高度,过高插不进去,过低,板插进去发光二极管露不出来。焊集成电路 LM339 时,为避免损坏集成块,建议将烙铁烧热后,马上从插座上拔下,利用烙铁的余热来焊接。最后将两个鱼尾夹焊在红、白两根导线上,并将导线的另一端由笔帽穿入,打一个结,再分别将红、白两根导线焊在(图 4 中)标有+、-的两个焊点上。将焊好的印刷电路板小心装入笔筒,将笔组装好。

调试:电路比较简单,只要焊接认真,基本上能一次成功。将焊好的逻辑笔红线接+5 伏,白线接电源地,此时黄色发光二极管点亮,表示测试点为高阻状态。当用探针轻触电源地时,绿色发光二极管点亮,表示测试点为低电平。当用探针接+5 伏时,红色发光二极管点亮,表示测试点为高电平。测试点为脉冲时,红、绿发光二极管交替点亮。

(本文介绍的普及型逻辑笔套件 15 元/套,邮费 2 元,邮编 100045,联系人:张文奇 庞文姬)

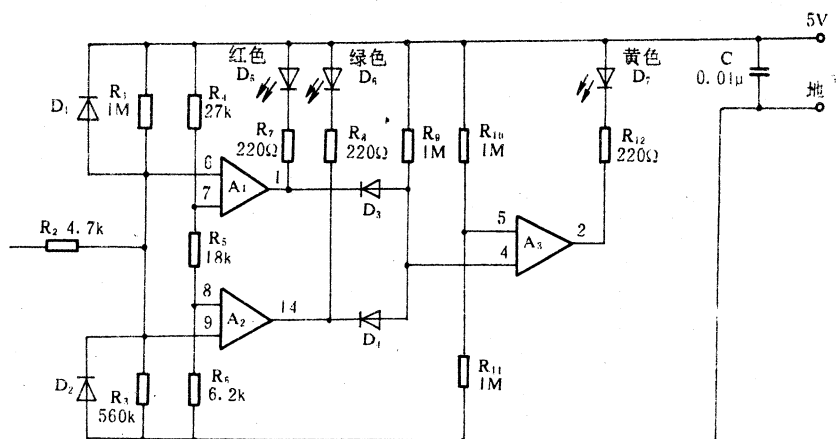


图 1

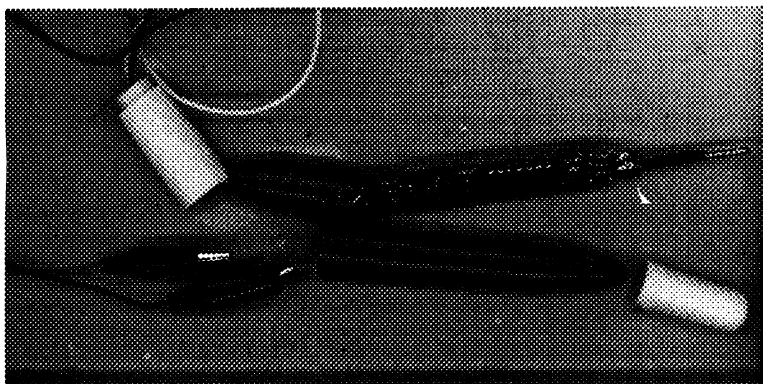


图 2

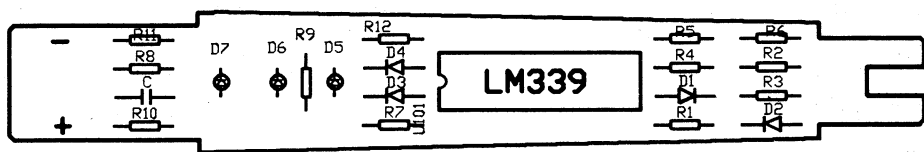


图 3

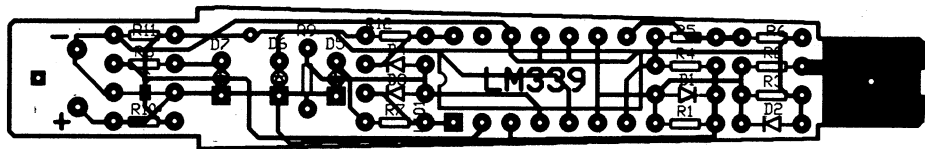


图 4

一种扩充单片机程序存储器的巧妙方法

王仕仲 湖南邵阳计算机厂(422000)

我厂批量生产的FR-4087袖珍计息机专用于银行进行储蓄、贷款利息计算。其CPU采用MC146805E2 CMOS单片机,共有13根地址线,可对8K外接存储器寻址,但是随着银行多次进行利率调整及计息方法的改变,程序量不断增大,超过8KB。于是需要在单片机外扩展其EPROM空间,下面具体介绍其实现方法。其中,经过扩展的EPROM之间的软件编程是此法的关键技巧。

一、硬件设计

1. 单片机MC146805E2的简单介绍。

单片机MC146805E2引出脚配置如图1所示。 $PA_0 \sim PA_7, PB_0 \sim PB_7$ 为输入/输出端口,它的每一根线可在软件控制下单独定义为输入或输出。8KB的空

间是由13根地址线决定的,它由 $B_0 \sim B_7$ 提供低8位, $A_8 \sim A_{12}$ 提供高5位。

2. 扩展单片机程序存储器线路设计

要扩大程序存储器空间,只有增加地址线,每增加一根地址线,空间扩大一倍。于是采用物理上增加空间的方法,即将口线 PA_0 当地址线使用,用 PA_0 位的“0”、“1”两种状态分别选中两个8KB区EPROM,使单片机外EPROM空间扩充到16KB。具体线路如图2所示, PA_0 为“0”状态选中第一区EPROM程序存储器, PA_0 为“1”状态则选中第二区EPROM程序存储器。

二、软件编程

1. 编程的关键

EPROM 的 PC 指示的地址处,假若先设置清位 PA。的指令,而 CPU 执行完该指令后,PC 值加 2 便转到第一区的 EPROM 的 PC 指示的地址处,这样整个程序杂乱无章,程序调试起来相当困难,更谈不上模块化、结构化设计,因而也无法确保程序的正确性。因此,编程时如何实现两区 EPROM 中程序连接便是一个关键问题了。

这里仍以第一区 EPROM 中在地址 0800H 去实现调用第二区 EPROM 入口地址为 0800H 的子程序为例,两区 EPROM 的编程如图 3 所示。下面给出 CPU 从第一区 EPROM 的地址为 0800H 开始执行程序的工作过程。

第二步: CPU 取第一区 EPROM 的 1C00~1C01H 的置位指令执行。指令执行后 PC 值变为 1C02H, PA₀ 变为高电平。

第三步:因 PA₀ 变高,CPU 选中第二区 EPROM,根据 PC 计数器的指示,取第二区 EPROM 的 1C02H~1C04H 的跳转指令执行。执行完该指令后,PC 值变为 0800H。

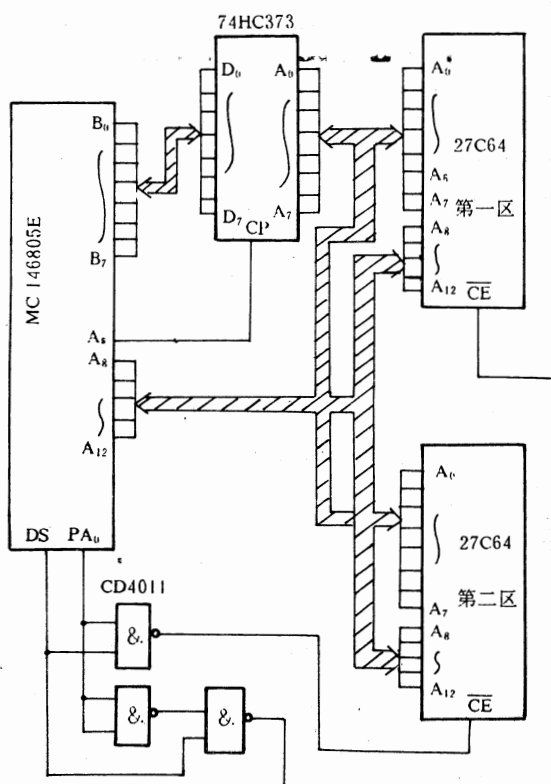


图 2

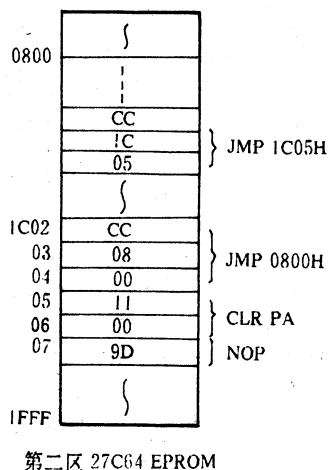
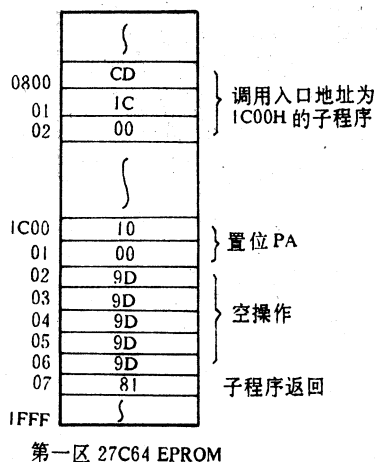


图 3

第四步：于是程序便从第二区 EPROM 的 0800H 处开始顺序执行程序，一直执行到该子程序的尾部，在该子程序的尾部设置一条跳转指令 JMP 1C05H，执行完该指令后，PC 值变为 1C05H。

第五步：CPU 取第二区 EPROM 的 1C05 ~ 1C06H 的指令执行。执行完该清 PA₀ 位的指令后，PA₀ 变为低电平，PC 值变为 1C07H。

第六步：由于 PA₀ 变低，CPU 便又选中第一区的 EPROM，从第一区的 EPROM 的 1C07H 取指执行。该指令是一条子程序返回指令，指令执行后，便从堆栈弹出两字节给 PC 计数器，因此 PC 值变为 0803H，于是整个程序实现了正常的返回，以后程序便又从第一区 EPROM 的 0803H 处开始顺序执行。

类似地可实现程序从第一区 EPROM 的一个断点处跳转到第二区 EPROM 执行相应的一段程序又转到第一区 EPROM 的断点处继续执行程序。

综上所述，实现程序在两区 EPROM 之间的相互调用、跳转的设计原理是：编程时进行统一安排，利用在同一区 EPROM 内采用扩充寻址的方法可实现程序的任意跳转或调用的特点，在两区 EPROM 的高地址区域预留足够的一些空间（上例从 1C00H 处开始，两区 EPROM 各留最后 1KB），以供子程序调用、返回或供程序修改时跳进、跳出的入/出口地址。这样便保证了两区 EPROM 内程序设计时空间使用的连续完整性，也便于实现程序的模块化、结构化设计，因而也能有效地保证程序运行的正确性。

人民邮电出版社最新出版软件

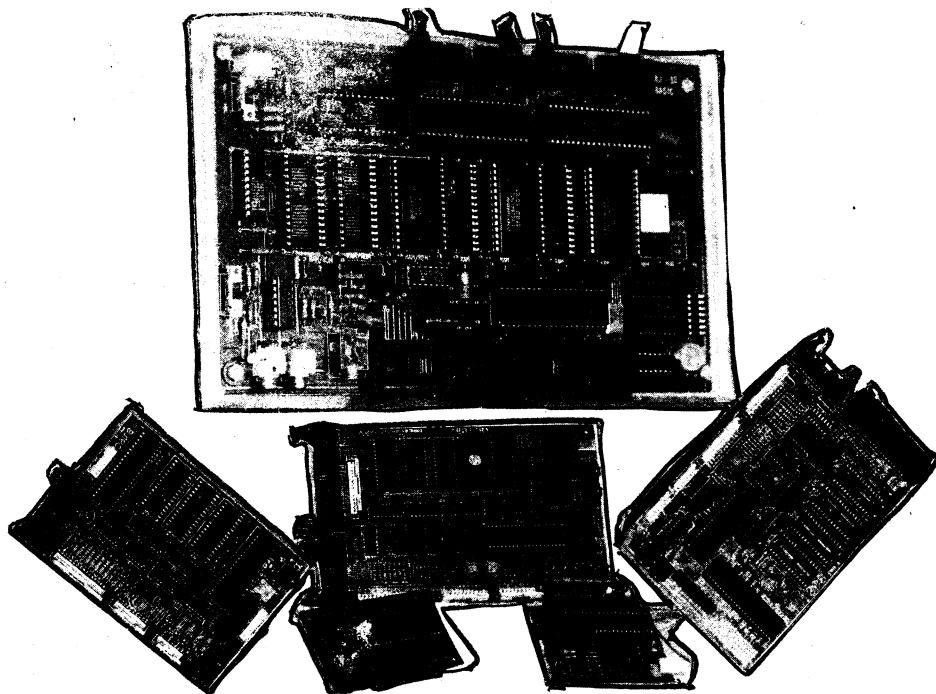
1. 计算机病毒清除工具 KILL 本软件系公安部计算机病毒研究人员经过七年的不懈努力推出的专用清病毒软件。可清除国内已发现的各种病毒。具有安全可靠、使用方便等特点，并保证软件的免费升级。定价：5 英寸低密盘一张 230 元/套；3 英寸低密盘一张 260 元/套。

2. 思成办公组合软件(原 GK 文字处理系统) 集编、排、打印于一体。具有多窗口编辑、“所见即所得”的排版、打印一键控制等功能，四种字体、字号无级变倍。适用于多种机型，且可被其它任何系统调用。软件配有一本操作手册。定价：5 英寸高密盘 7 张 600 元/套；3 英寸高密盘 7 张 720 元/套。

3. 严氏数据库 2.00 版(MTDBASE2.00)

采用 C 语言编程，全部操作均以菜单形式进行，不仅具有数据库描述、数据库操作等基本功能，还具有多层表头的全屏幕编辑，备注型字段的输入、输出、修改一键操作，汉字数据资料的缺省查询，字符型与数值型资料自动组合查询等功能。软件配有一本操作手册。定价：5 英寸低密盘两张 360 元/套。

欲购者请与我们联系，地址：北京市东长安街 27 号人民邮电出版社电子图书编辑部 (100740) 联系电话：5130985 联系人：高乐忠



特别介绍:BJ52 自开发系统

- 系统备有 BASIC 解释程序,用户可用 BASIC 高级语言编程
- 备有 8K 汇编级监控程序
- 在系统上可方便地固化 E²PROM 芯片
- BJ52 是物美价廉实用的应用板,售价 998 元

春天到 学电脑 兴科技 掀高潮

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。它以多种方式提供给用户。该系统适用于初学者,中小学生,工矿企业科技人员及其他电脑爱好者。

- DP-851 整机,售价 438 元
- 单板型自开发器 A 型套件(主板+PC 机电缆)售价 279 元/套
- 自开发器 B 型套件(主板+薄膜键盘板)售价 324 元/套
- 教学实验板配置(自开发器+实验板+辅助件)售价 397.5 元/套
- 另备有逻辑笔、固化板、三位半 A/D、四位半 A/D、A/D D/A 板等多种扩展板。

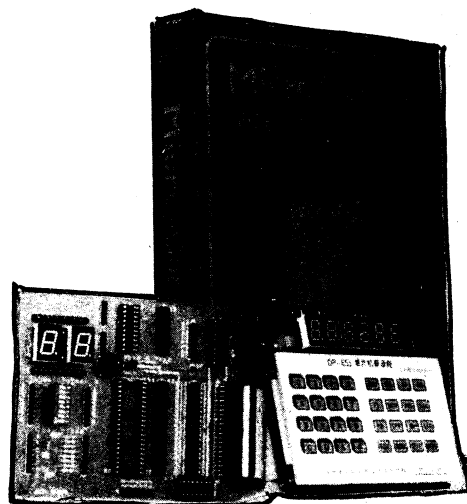
开户行:工商银行北京百万庄分理处

帐号:014-047014-96

联系人:张文奇 庞文姬

地址:北京西城区月坛南街 83 号

邮编:100045 电话:8512947



8086 接口实验系统设计

江琪 北京电信学校(100021)

伴随着 Intel 8088/8086 以及 80286、80386 等 CPU 芯片日益广泛深入的应用,与之相配套的众多可编程的接口芯片也得到了广泛应用。就目前国内院校的情况来看,已陆续开始开设 16 位微机原理及接口控制等课程,但都普遍存在着接口实验不容易解决这样一个问题。为此,我设计了 8086 接口实验卡,通过它可以很好地解决上述问题,使理论教学与实验能够紧密地结合起来。

一、国内 8086 接口实验现状

就国内 16 位微机原理课程教材而言,目前主要以讲授 8088/8086 为主,与之相配套的可编程接口芯片包括:8255,8253,8251,8259 等芯片,而完成这些接口芯片的实验可有三种方法:

1. 借助 IBM-PC 机完成

IBM-PC 机(包括 PC/XT,AT 及兼容机)都含有上述的接口芯片,借助于系统机可以完成一些实验,但是它具有以下几点不足:

- ①不能完成所有的实验
- ②实验不直观
- ③不能使学生充分动手

2. 借助于单板机完成

使用 Intel 公司的 SDK-86 单板机或国产的 TP-86 单板机能够完成上述的接口实验,但就目前国内院校的条件而言,它有以下几点不足:

- ①增加了大量实验经费,需另外配置价格较昂贵的数十台单板机
- ②需专门开辟一实验室
- ③需另外掌握一套单板机的操作方法,且相对操作复杂
- ④显示不直观

3. 借助于接口实验板完成

设计一个接口实验系统,该系统充分利用 PC 机的资源,这样就可以借助于 PC 机对接口实验板编程,从而达到完成接口实验的目的,这种实验方法的优点是:

- ①能够充分利用 PC 机的资源
- ②操作简便
- ③成本低廉而又不必另建实验室
- ④实验结果直观,学生可以充分动手
- ⑤能够完成所有接口实验

根据邮电部中专“十六位微机原理”教学大纲中对接口实验的要求以及各校的实际情况,结合现在主流机 IBM-PC 系列机的硬件特点,本着以下几个原则

对实验卡进行设计的:

- 1. 完成大纲要求的所有接口实验,包括:8259,8255,8253,8251。
- 2. 在各种 IBM-PC 系列机上(PC,PC/XT,286,386 等)均能工作。
- 3. 尽可能加强学生的动手能力。
- 4. 实验结果直观,明确,宜于教师检查。
- 5. 实验卡的硬件设计简练,实用,易于掌握。

整个实验系统由接口卡和实验板这两部分组成,接口卡完成 PC 机与实验板的硬件连接,实验板供学生实验使用。

二、接口卡设计

接口卡由地址译码器电路、数据总线缓冲器和时钟发生器电路组成(接口卡原理图见附录 1)。

1. 地址译码器

由 IBM-PC 的端口地址分配可知,IBM-PC 机系列的所有机器中,地址 100H—1EFH 段内没有安排扩展卡,故我们选用这个地址范围来作为我们的接口卡各 I/O 端口地址,所以 74LS138 的六个输入端接法如图 1 所示。

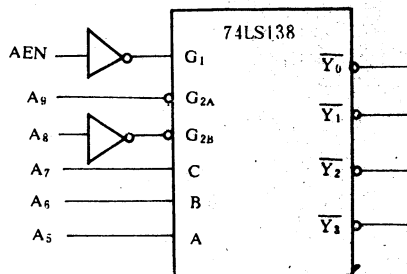


图 1

这里 A9, A8, A7, A6, A5, AEN 与译码器输出端的关系如下表所示。

AEN	A9	A8	A7	A6	A5	译码输出端
1	0	1	0	0	0	$\overline{Y_0}$
1	0	1	0	0	1	$\overline{Y_1}$
1	0	1	0	1	0	$\overline{Y_2}$
1	0	1	0	1	1	$\overline{Y_3}$

74LS245 芯片用作系统机与实验板之间的总线发送/接收器, $G=0$ 时, 74LS245 工作。此时, 若 $DIR=0$,

则 245 为 B→A 方向传输;若 DIR=1,则为 A→B 方向传输,我们将实验卡使用的 $\overline{Y0}$, $\overline{Y1}$, $\overline{Y2}$ 和 $\overline{Y3}$ (74LS138 的输出)作为 74LS21(四输入与门)的输入端,输出端接往 245 的 G 端,这样当地址选用 $\overline{Y0}$, $\overline{Y1}$, $\overline{Y2}$ 和 $\overline{Y3}$ 时(或的关系),21 的输出为低电平,则 245 芯片工作,否则,当 $\overline{Y0}$, $\overline{Y1}$, $\overline{Y2}$ 和 $\overline{Y3}$ 均为高电平时,245 芯片不工作,则实验板上的各芯片均不会与系统机相连,将 DIR(245)接往 PC 机插槽上的 IOR,即可满足 PC 机对实验板上各芯片的读写要求。

2. CLK 时钟发生电路

8284 的 X1,X2 端接有一个 14.318MHz 的晶体,它的 6 分频输出端 PCLK 输出 2.386MHz 的方波,又经过 2 分频(74LS175),产生一个 1.19MHz 的方波以供给实验板上的 8253A 芯片使用。

三、实验板的硬件设计

原理图见附录 2。由图可见,实验板由以下几部分组成:

1. 8253

在此,只对关键几个引脚做出如下说明:

①CLK 引脚:对于 CLK0,CLK1 和 CLK2,我们均把它们接往接口卡上的 8284 的 1.19MHz 频率输出端(这个频率值与 PC 机中 8253 的值相同)。

②GATE 引脚:我们将 GATE0,GATE1 和 GATE2 均接往高电平,全部打开计数/定时器。但考虑有些实验的实际需要,我们分别从这三个引脚引出了焊点(插座)可供使用,如图 2 所示的 A,B,C 点。

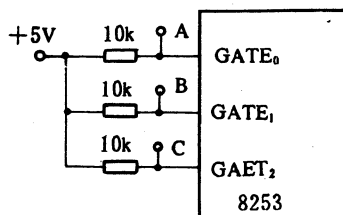


图 2

这样如需要把 GATE0 由其它电路控制。可将控制引线接往焊点 A。

③8253 的 \overline{CS} 引脚接至接口卡中 74LS138 的 $\overline{Y1}$,使当 8086/8088 的地址线 A9,A8,A7,A6,A5 为 01001 时选中该芯片。即 8253 地址范围是 120H~13FH。

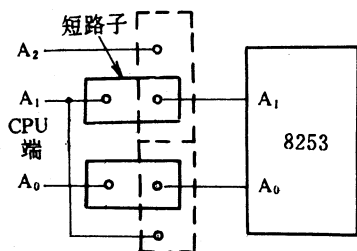


图 3

④对于 8253 的 A1,A0 引脚,我们做了如图 3 所示的处理:

这样当把两个短路子横向插时,则 8253 的 A1,A0 分别接往 CPU 的 A1 和 A0,这时 8253 端口地址如下表所示:

A9 A8 A7 A6 A5 A4 A3 A2 A1 A0	选用地址	端口名称
0 1 0 0 1 X X X 0 0	120H	计数器 0 端口
0 1 0 0 1 X X X 0 1	121H	计数器 1 端口
0 1 0 0 1 X X X 1 0	122H	计数器 2 端口
0 1 0 0 1 X X X 1 1	123H	控制端口

当把两个短路子按图示虚框纵向插入时,则 8253 的 A1,A0 分别接往 CPU 的 A2,A1,这时 8253 的端口地址如下表所示:

A9 A8 A7 A6 A5 A4 A3 A2 A1 A0	选用地址	端口名称
0 1 0 0 1 X X 0 0 X	120H	计数器 0 端口
0 1 0 0 1 X X 0 1 X	122H	计数器 1 端口
0 1 0 0 1 X X 1 0 X	124H	计数器 2 端口
0 1 0 0 1 X X 1 1 X	126H	控制端口

⑤8253 的三个输出引脚均悬空,接至实验板插子的第 3,4,5 脚,由学生自己连线使用(可接至中断请求 IRQ4 端)。

2. 8251A

8251A 用来进行通信实验,在硬件设计上将它的 TXD 和 RXD 经过 MC1488 和 MC1489 的电平转换而与外面联系。它的 \overline{CS} 信号来自接口卡上 74LS138 的 $\overline{Y2}$,而它的 C/D 信号来自 CPU 的地址线 A1,故有下表所示的操作。

A9 A8 A7 A6 A5 A4 A3 A2 A1 A0	选用地址	\overline{RD}	\overline{WR}	对应的操作
0 1 0 1 0 X X X 0 X	140H	0	1	CPU 从 8251A 输入数据
0 1 0 1 0 X X X 0 X	140H	1	0	CPU 往 8251A 输出数据
0 1 0 1 0 X X X 1 X	142H	0	1	CPU 读取 8251A 的状态
0 1 0 1 0 X X X 1 X	142H	1	0	往 8251 写入控制命令

CLK 端引自接口板的 1.19MHz 输出端,TXRDY 和 RXRDY 经过或门而接至 PC 机的 IRQ3,这样只要在软件上修改了 0BH 号中断类型(PC 机用于 COM1)的中断向量,即可完成 8251A 中断方式下的串行通信功能。

3. 8255A

8255A 具有三个可编程的并行口。这三个口可分别安排输入和输出等几种工作方式。在实验板的设计上,我们安排 PA,PB 为输出口,用来控制 8 位七段显示器 LED 的显示;安排 PC 口为输入口,用来监视按键的工作,而上面这种安排是靠软件来支持的(由学生自己编制),也可以由学生改变这种安排(通过拔下板上的部分短路子来完成),下面分别介绍:

①8255A 的端口地址

8255A 的 CS 信号引至接口卡上 74LS138 的 $\overline{Y_0}$, A1, A0 可通过选择引至 CPU 的 A1, A0 或 A2, A1 (方法同 8253 芯片介绍④), 当 8255A 的 A1, A0 连到 CPU 的 A1, A0 时, 8255A 各端口地址如下表所示:

A9	A8	A7	A6	A5	A4	A3	A2	A1	A0	选用地址	端口名称
0	1	0	0	0	X	X	X	0	0	100H	端口 A
0	1	0	0	0	X	X	X	0	1	101H	端口 B
0	1	0	0	0	X	X	X	1	0	102H	端口 C
0	1	0	0	0	X	X	X	1	1	103H	控制端口

当 8255A 的 A1, A0 连到 CPU 的 A2, A1 时, 8255A 端口 A, 端口 B, 端口 C 和控制口的地址分别选用: 100H, 102H, 104H 和 106H。

②显示电路

实验板上采用 8 个共阴极的 LED, 它的各段选位均连接在一起由 PA 口来控制。考虑到驱动问题, 不是将 PA 口的输出直接接往段选端, 而是经过了 74LS06 芯片 (集电极开路非门) 以后接往段选端的, 如图 4 所示。

现若想要某数码管显示数字 5, 则由 PA 口应送出 68H, 即 01101000B, 经过非门后变为:

c Dp e d b a f g 即如图
1 0 0 1 0 1 1 1

这样可得到数字 0—F 的字段码值, 如表所示。

图 4 中的 J2 为八个短路子, 当插上这八个短路子时, PA7—PA0 就接往段选端, 当按下这八个短路子时, 则 PA7—PA0 就处于悬空状态, 学生可通过 J1 而使用 PA 口。

8255A 的 PB 口用来控制 LED 的位选端, 同样考虑到驱动问题, PB7—PB0 也不是直接接到各个数码管的位选端, 而是经过了 75452 驱动电路 (非门)。

在这里, 我们也同样考虑到了学生上机的主动性, 而设计了如上述的 J1 和 J2, 使 PB 口悬空, 如果插上

J2, 则 PB0 选择最高位数码管, ……PB7 选择最低位数码管。

②键盘电路

在 PC 口的设计上我们同样考虑了 J1, J2, 如果插上 J2, 则 PC 口的八个引脚就连到了八个按键, 我们安排 PC7 连到键 7, PC6 连到键 6, ……PC0 连到键 0, 这样当无键按下时, PC7—PC0 均为高电平输入; 当有键按下时, 则相应位为低电平输入, 其余位为高电平输入。

4. 中断实验

通过 8259A 中断控制器可以很方便地扩展出八个 INTR 中断源。考虑到 PC 机的特点, 我们没有另外设计一片 8259A 芯片, 而是直接利用 PC 机 62 芯总线插槽提供的 IRQ3—IRQ4 来完成中断实验, 这样一方面降低了实验板的成本, 另一方面也完成了大纲的要求。

PC 机 8259A 与 8088 的连接示意图如图 5。

在 PC 机中, 8259A 中的八位中断请求输入端中的 IRQ0 接计数/定时器 8253 通道 0 的输出端, 作为系统计时时钟的请求输入端, IRQ1 接收键盘接口电路送来的请求信号, 286 机的 IRQ2 接往另一片 8259A 芯片, 剩下的 5 个 (PC 机为 6 个) 请求端接到 62 芯总线上, 用来接收扩展卡发出的中断请求。

8259A 的片选 \overline{CS} 端接板上地址译码器 74LS138 的 \overline{INTRCS} 输出端, 偶地址端口为 20H, 奇地址端口为 21H, 在 BIOS 中, 通过 ICW1—ICW4 对 8259A 的工作方式设置如下:

采用全嵌套方式, IRQ0 的级别最高, IRQ7 的级别最低, ICW2 为 80H, 这样八个中断类型码为 08H—0FH; 中断请求信号采用边沿触发方式; 用普通的中断结束方式, 即在中断处理子程序末尾需用中断结束命令; 用普通屏蔽方式。

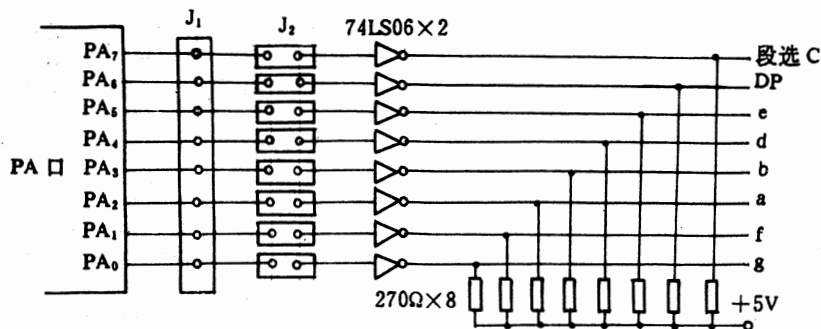


图 4

a	0	41
f g b	1	77
e c, dp	2	C2
d	3	62
	4	74
	5	68
	6	48
	7	73
	8	40
	9	60
	A	50
	B	4C
	C	C9
	D	46
	E	C8
	F	D8

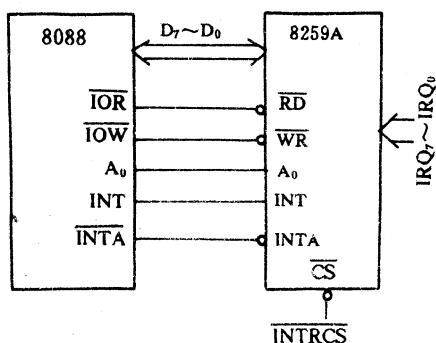


图 5

根据以上特点,我们将 8259A 的 IRQ3 和 IRQ4 接至实验板上,其中 8251A 占用了 IRQ3,IRQ4 悬空供学生实验用,而 IRQ3 的中断类型为 0BH,IRQ4 的中断类型为 0CH,学生可利用 DOS 功能调用 (INT 21H) 中的 25H、35H 调用来改变中断向量,进而达到编写自己的中断服务子程序的目的。

5. 其它设计

①接口卡与实验板的连接

实验板通过一个 26 芯的扁平电缆与接口卡连接,各引脚的名称如图 6 所示。

1. 19M	26	25	CLK
GND	24	23	+5V
-12V	22	21	+12V
A1	20	19	A2
RESET	18	17	A0
IOR	16	15	IOW
IRQ3	14	13	IRQ4
$\overline{Y2}$	12	11	$\overline{Y3}$
$\overline{Y0}$	10	9	$\overline{Y1}$
D6	8	7	D7
D4	6	5	D5
D2	4	3	D3
D0	2	1	D1

图 6

为了方便学生实验,我们将图中各引脚分别引出,可供实验选用。这里 CLK 输出 4.7MHz 的方波, $\overline{Y3}$ 片选信号实验板未用,地址范围是:160H—17FH。IRQ4 悬空可供使用。

②在实验板的最右端我们设计了一个九脚的插座,如图 7 所示,实验板使用了上面的八脚,第九脚(最下面一脚)没有用。这八脚均来自各芯片。

1	Vcc
2	GND
3	OUT0
4	OUT1
5	OUT2
6	TXD
7	RXD
8	RXCLK
9	NC

图 7

③为了方便实验,我们在实验板的最左端的上、下端安排了一个 16 脚的 IC 座和二 14 脚的 IC 座(Vcc 和 GND 均已连好)供实验用。比如某实验中需用到非门,则可把一片 74LS04 插入到一个 14 脚的 IC 座上,通过插线(已留出插座)而完成连接。

另外在实验板上我们还安排了二排插座,供实验备用,如图 8 所示:

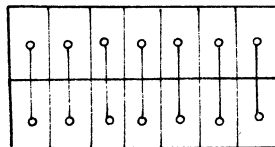


图 8

四、实验举例

1. 8255 实验

要求:①完成硬件连接,即 8255 端口 A 工作于输出方式,用来输出 LED 段选值;端口 B 也工作于输出方式,用来输出 LED 位选值;端口 C 为输入方式,用来输入按键的状态。

②八只按键分别代表数字 1,2,...,8,当按下某一键时,该键所代表的值则在 LED 上自左向右循环显示,而又按下另一键时,则换字显示,同时在 LED 上显示的数字会在 CRT 上显示。

程序框图及程序清单如下:

几点说明:

①程序使用 Turbo ASM 汇编完成。

②8255A 的控制字是 89H,即端口 A 输出,端口 B 输出,端口 C 输入。

③ neg al

dec al

这两条指令完成求反功能。

```
.model small
.stack 200h
.data
wd db 0b7h,0c2h,92h,0b4h ;display segment values
db 98h,88h,0b3h,80h
```

```

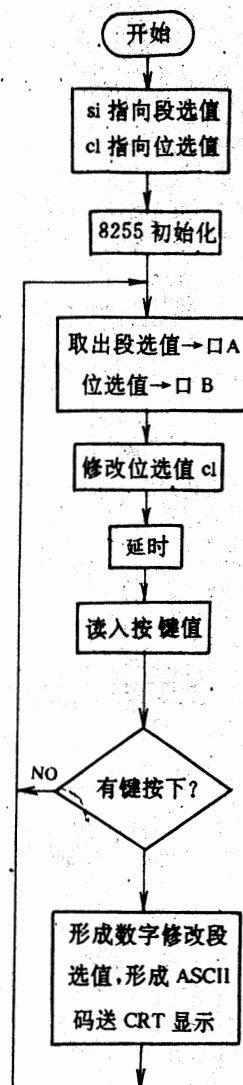
.code
mov ax,@data
mov ds,ax
mov cl,01h           ;move display bit value to cl
mov si,offset wd
mov bx,7
mov al,89h           ;8255 control word
mov dx,106h          ;8255 control port address
out dx,al
jia: mov al,[si+bx-1] ;get display segment value
mov dx,100h          ;port A address
out dx,al
mov dx,102h          ;port B address
mov al,cl
out dx,al
rol cl,1             ;for next display bit
push bx
mov bx,0005h         ;delay
nnn1: mov ax,00
nnn: dec ax
and ax,ax
jnz nnn
dec bx
jnz nnn1
pop bx
mov dx,104h          ;port C address
in al,dx             ;read in key value
neg al
dec al
and al,al
jz jia               ;no key press, jump to jia
mov bl,0             ;have a key press
abc1: or al,al
jz jia
inc bl               ;count key value to bl
shr al,1
jnc abc1
mov al,bl            ;display key value at CRT
and al,0fh
mov bl,al
or al,30h
mov dl,al
mov ah,02h
int 21h
mov bh,0             ;there is key value in bx
jmp jia
end

```

```

.code
mov ax,0004h
int 10h
mov bx,0007h
mov ah,0bh
int 10h
mov al,89h           ;8255 control word
mov dx,106h          ;8255 control port address
out dx,al
mov al,36h           ;have a key press, write 8253cw
mov dx,126h          ;8253 control port
out dx,al
mov dx,120h          ;8253 counter0 address
mov al,0ffh          ;write low 8 bits
out dx,al
mov al,35h           ;write high 8 bits

```



2. 8253 实验

要求: ①完成硬件连接: 打开 GATE0(高电平), 将 OUT 接往 8255A 的 PC7

②8253 工作在方式 3, 即方波输出

程序框图及清单如下:

```

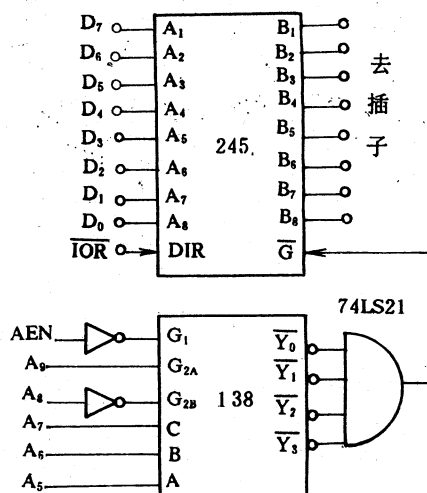
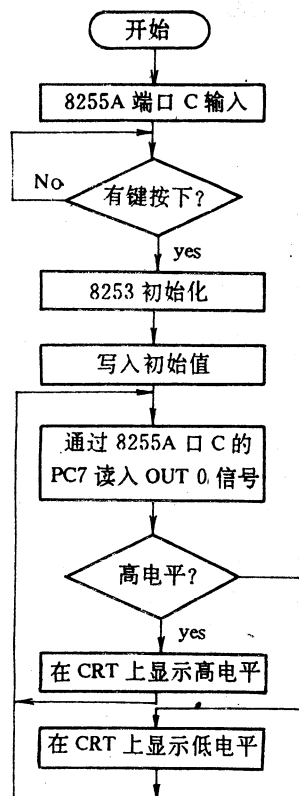
.model small
.stack 200h
.data

```

```

out dx,al
dd0:
mov dx,104h
in al,dx          ;read in CLK0
neg al
dec al
mov dx,0070
and al,80h        ;test CLK0 level
jnz dd1           ;low level
add dl,20         ;high level
dd1:
mov al,02h
mov ah,0ch
int 10h
inc cx
cmp cx,319
jz dd10
mov bx,05fh
wait2: dec bx
jnz wait2
ddd: jmp dd0
dd10:
mov ah,6h
mov al,0
mov bh,0
mov ch,0
mov cl,0
mov dh,24
mov dl,79
int 10h
jmp dd0
end

```



接口卡件原理图

敬告各界

广大读者、软件开发人员和用户
若有软件出版、推广、培训和咨询事宜,请与中国软件行业协会软件出版
分会办公室联系。

通信地址:北京 173 信箱(普惠兆
里 12 号)

软件出版分会

邮 码:100036

开户银行:北京市万寿路信用社

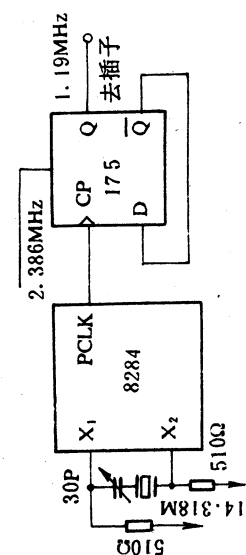
户 名:中国软件行业协会软
件出版分会

帐 号:07203

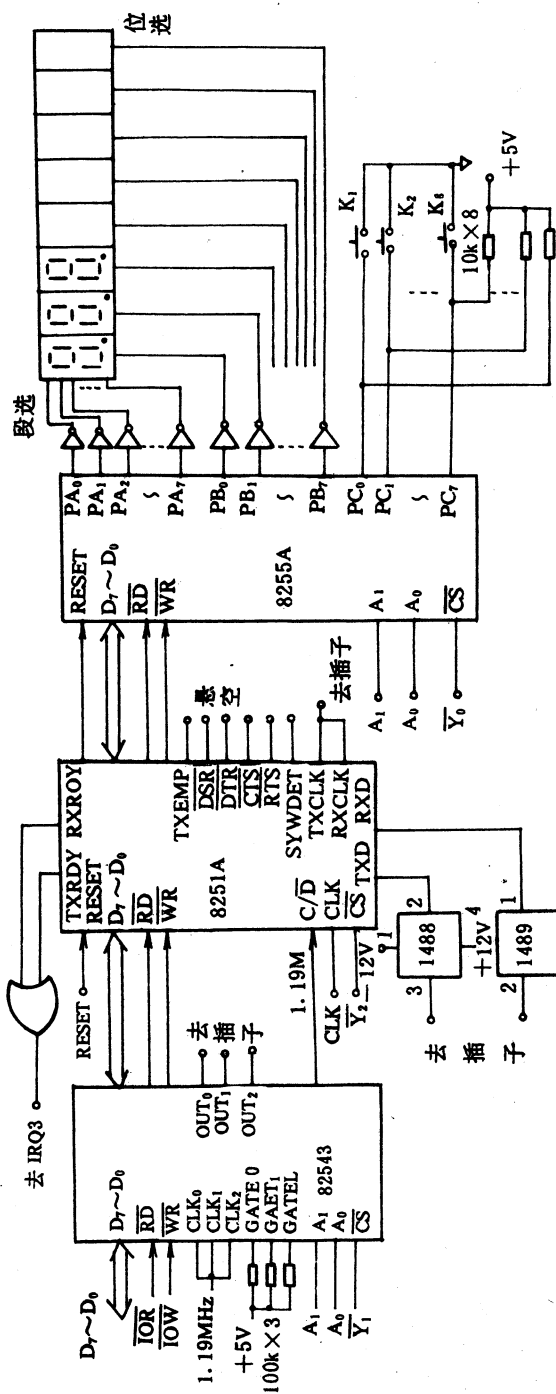
联 系 人:李 涛

电 话:8221677

接口卡硬件原理图



26	1. 19	CLK	25
24	GND	+5V	23
22	- 12V	+12V	21
20	A ₁	A ₂	19
18	RESET	A ₀	17
16	$\overline{\text{IOR}}$	$\overline{\text{IOW}}$	15
14	IRQ3	IRQ ₄	13
12	$\overline{\text{Y}}_2$	$\overline{\text{Y}}_3$	11
10	$\overline{\text{Y}}_4$	$\overline{\text{Y}}_1$	9
8	D ₈	D ₇	7
6	D ₆	D ₅	5
4	D ₂	D ₃	3
2	D ₀	D ₁	1



实验卡硬件原理图

PC 机多功能控制卡特殊故障维修一例

刘绍华 辽宁省朝阳市人民银行(122000)

东海 D 微机所配软盘控制卡是一种多功能控制卡,上面除了软盘适配器外还包括两个串行通信口、一个并行打印口、一个游戏口和一个实时时钟口。本例为软盘适配器口故障。

故障现象:从软盘引导系统时,屏幕显示 PARITY CHECK(1) 0000(S),系统死机;从硬盘引导时,系统启动正常,但当读写 A、B 盘时屏幕均显示 PARITY CHECK(1) 0000(S),系统死机。

故障分析与检修:因为从硬盘引导正常,证明不是系统内存故障。用一个好的软盘控制卡替换,故障排除,证明是原软盘控制卡的故障。

PARITY CHECK (1)是系统板内存奇偶校验出错信息,当出现奇偶校验错时系统将向 CPU 发一个非屏蔽中断,主机将处于停机状态。产生非屏蔽中断的条件是:系统板上的 RAM 在读写时产生奇偶校验错;I/O 通道扩展板上出现奇偶校验错;8087 协处理器有异常中断请求。本例中主机板上未装 8087,多功能卡上也没有扩展内存,所以肯定是系统板上的内存出现了奇偶校验错。因为奇偶校验错信号发生电路位于系统板上,而前面我们已证明系统板是好的,所以排除了奇偶校验错发生电路出故障的可能,很有可能写入内存的数据是杂乱无章的,当再读出来时,就产生了奇偶校验错。

比如执行 DIR A:↙,执行过程是:首先将 A 盘文件信息写入系统内存某处,再从该处读出写入 VRAM,在显示器上显示出来。当读内存时就产生了奇偶校验错。

为了能读写软盘,做如下操作:

C>DEBUG↙

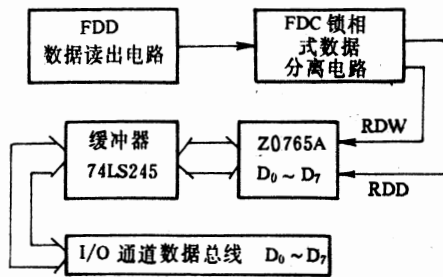
-O A0 00↙

-Q ↙

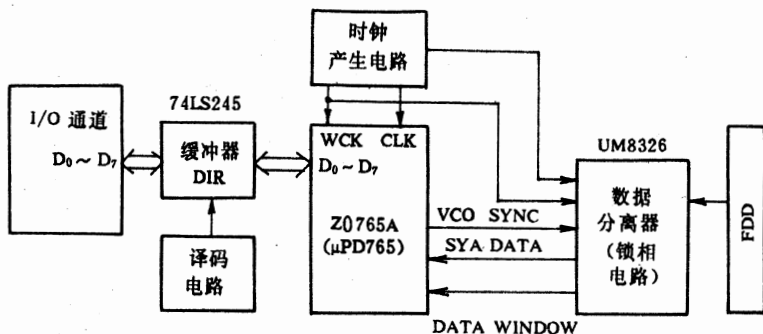
使 NMI 屏蔽触发器置 0,禁止向 CPU 发中断信号。

执行 DIR A:↙,屏幕上显示的是一堆不可辨认的字符或者是残缺不全的信息,证明确实是软盘控制卡读数据通道故障。再往 A 盘上拷一个文件,到别的机器上验证,证明写数据通道正常。

从软盘到主机的读出数据通道是:



更换一个 74LS245,故障依旧,检查锁相式数据分离电路发现该电路不正常,该集成电路型号为 UM8326,更换后,故障排除。见图。



图

第五章 背景画面的位移操作(上)

于春 山东苍山机械电子化学工业局(277700)

背景画面的位移一般分为横向位移、纵向位移和斜向位移三大类。通常称横向位移为画面的左右滚动、纵向位移为画面的上下滚动。多数 CPU 的位移控制操作是通过对屏幕显示映射单元不停地写入数据实现的。而 6527 CPU 则另辟蹊径,采用了屏幕显示映射单元的内容不变,通过移动屏幕显示窗口实现画面位移的方法。这犹如照相机或摄像机中的取景框,显示窗口对准了哪一块图形,就把这一块图形显示在屏幕上。因此,背景画面的位移,实质上是显示窗口向相反的方向移动。由于显示窗口可任意移动,因此 6527 CPU 可方便地进行任意方向的位移(一般取八个方向)。由于每一个方向的位移可分为往复、循环两种过程,所以 6527 CPU 一般有 16 种位移方式。

繁多的位移方式和简单的位移操作是 6527 CPU 的突出特点。

下面分别介绍各类位移操作的原理和实现方法。

一、屏幕显示窗口

1. 显示窗口的概念

屏幕显示窗口是一个形象的称谓,实际上它是由卡通零页、卡通一页、底背景页三层画面叠加而成的一个具有立体结构的画面。因此,显示窗口的平面尺寸为 32 列×30 行。为方便讨论,我们把它抽象为一个具有 32 列×30 行的透明矩形,在示意图中以双线框表示,讨论中简称显示窗口或窗口。

虽然画面位移的实质是显示窗口移动,但为了符合画面位移的通常概念和称谓习惯,我们在讨论中仍称背景画面位移。

2. 显示窗口的坐标系

为使显示窗口的座标与卡通显示座标相对应,我们把窗口的座标原点选在左上角,如图 1 所示。座标单位按点阵计算,X 取值范围为 0~255,Y 取值范围为 0~239。

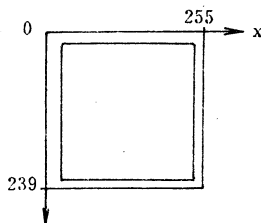


图 1 显示窗口坐标系

3. 显示窗口与背景画面的对应关系

由于画面的移动,实质上是窗口向相反方向的位移,那么对于图 1 选定的窗口坐标系,背景画面与显示窗口的对应关系将如图 2 所示。

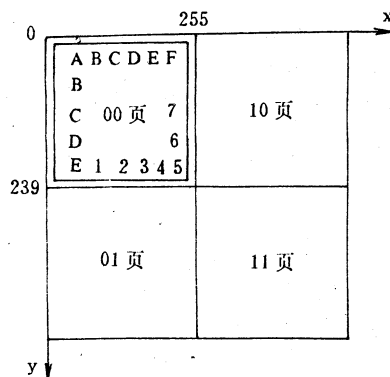


图 2 显示窗口与背景画面的对应关系示意图

二、画面的简单位移

所谓画面的简单位移是指把窗口移到画面的某个固定位置的操作。

1. 画面的横向位移

画面横向位移的示范程序如 No. 5-1(以移动 a 点为例)。

No. 5-1

```
LDA # $a      ;a=0~FF
STA $2005     ;窗口横向位移 a 点
LDA # $00
STA $2005     ;窗口纵向位移 0 点
JSR YANSHI
RTS
YANSHI LDX # $00 ;延时子程序
X1     INX
```

```
NOP
NOP
NOP
NOP
BNE X1
INY
BNE X1
RTS
```

执行程序 No. 5-1 将有图 3 所示的画面。

程序 No. 5-1 对应的 F BASIC 程序如 No. 5-2。

```
5 REM No. 5-2
10 POKE 8197,a
20 POKE 8197,0
```

30 PAUSE 1000;END

2. 画面的纵向位移

若令画面纵向位移 b 点, 其实现程序如 No. 5-3.

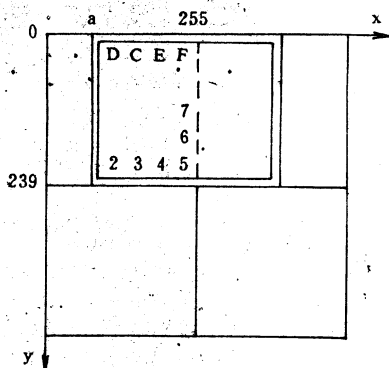


图 3 画面横向移动 a 点示意图

No. 5-3

```
LDA # $00
STA $2005
LDA # $b
STA $2005
JSR YANSHI
RTS
```

对应的 F BASIC 程序如 No. 5-4

5 REM No. 5-4

10 POKE 8197,0

20 POKE 8197,b

30 PAUSE 1000;END

执行 No. 5-4, 将有图 4 所示的画面。

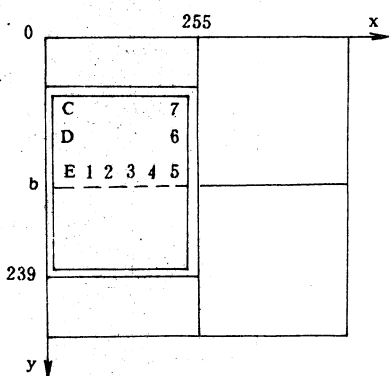


图 4 画面纵向移动 b 点示意图

3. 画面的斜向位移

若令画面横向位移 c 点、纵向位移 d 点, 其实现程序如 No. 5-5

No. 5-5

```
LDA # $c
STA $2005
LDA # $d
STA $2005
```

JSR YANSHI

RTS

其对应的 F BASIC 程序如 No. 5-6.

5 REM No. 5-6

10 POKE 8197,c

20 POKE 8197,d

30 PAUSE 1000;END

可以看出, 画面的斜向位移就是画面横向位移和纵向位移的共同(合成)作用结果。

斜向位移示意图如图 5.

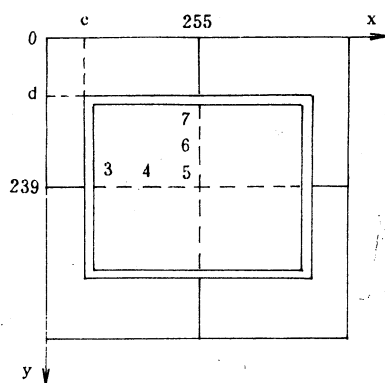


图 5 画面斜向位移示意图

通过以上的讨论可见, 6527 CPU 工作系统的画面位移操作是极为简单的。

三、画面的连续位移

实际游戏中, 画面的位移操作多数是以点为单位的连续位移。如《碰碰车》游戏是连续的画面横向位移;《拆屋工》则是连续的画面纵向位移;《淘金者》、《沙罗曼蛇》等游戏则是画面横向、纵向连续分段位移;而大多数游戏如《魂斗罗》、《松鼠大战》、《上尉密令》、《人间兵器》等游戏则是横向、纵向混合位移, 从而使游戏画面广阔博大、丰富迷人。

一般把画面的横向连续位移称为横向滚动, 把画面的纵向连续位移称为纵向滚动。下面讨论各类连续位移的实现方法。

1. 画面连续左移

从图 5-3 可以看出, 画面的横向位移是在 00、10 页或 01、11 页显示页面间的移动。根据程序 No. 5-1 我们很容易写出画面连续左移的程序 No. 5-7

No. 5-7 LEFT MOVE

LDX # \$00

LDY # \$00

LEFT STX \$2005

STY \$2005

JSR YANSHI

INX

BNE LEFT

RTS

对应的 F BASIC 程序见 No. 5-8

```
5 REM No. 5-8 LEFT MOVE
10 A=0
20 POKE 8197,A
30 POKE 8197,0
40 A=A+1;IF A=256 THEN 70
50 PAUSE 5
60 GOTO 20
70 PAUSE 1000;END
```

执行程序 No. 5-8 可把画面的 00 页连续左移出窗口,把 10 页连续左移入窗口,使窗口对正 10 页。

调整 50 行 PAUSE 语句的参数,可控制画面左移的速度。

2. 画面的连续右移

画面的连续右移程序见 No. 5-9, F BASIC 程序见 No. 5-10。

```
No. 5-9 RIGHT MOVE
LDX # $FF
LDY # $00
RIGHT STX $2005
STY $2005
JSR YANSHI
DEX
BNE RIGHT
RTS
5 REM No. 5-10 RIGHT MOVE
10 A=255
20 POKE 8197,A
30 POKE 8197,0
40 A=A-1;IF A=0 THEN END
50 PAUSE 5
60 GOTO 20
```

画面连续右移时,首先把显示窗口迅速移到 10 页面,然后连续右移,直到把 00 页全部移入显示窗口而停止。

3. 画面的连续上移

由图 5-4 可见,画面的纵向位移是在 00、01 页或 10、11 页面间的移动。画面连续上移的实现程序见 No. 5-11、No. 5-12。

```
No. 5-11 UP MOVE
LDX # $00
LDY # $00
UP STX $2005
STY $2005
JSR YANSHI
INY
CPY # $F0
BNE UP
RTS
5 REM No. 5-12 UP MOVE
10 A=0
20 POKE 8197,0
30 POKE 8197,A
40 A=A+1;IF A=240 THEN 70
```

```
50 PAUSE 2
60 GOTO 20
70 PAUSE 1000;END
```

运行程序 No. 5-12, 00 页将逐渐上移出显示窗口, 01 页则逐渐移入窗口。

4. 画面的连续下移

画面的连续下移程序见 No. 5-13, No. 5-14。

No. 5-13 DOWN MOVE

```
LDX # $00
LDY # $F0
DOWN STX $2005
STY $2005
JSR YANSHI
DEY
BNE DOWN
JSR YANSHI
RTS
5 REM No. 5-14 DOWN MOVE
10 A=240
20 POKE 8197,0
30 POKE 8197,A
40 A=A-1;IF A=-1 THEN END
50 PAUSE 2
60 GOTO 20
```

同画面的连续右移情况相同,运行程序 No. 5-14 后,显示窗口要迅速跳到 01 页,然后逐点上移到 00 页,直到窗口与 00 页对正后,位移过程结束。

5. 画面的横向往复位移

《淘金者》I 代游戏中,按暂停键后,可使用操纵器的左、右方向按钮控制画面左、右往复运动。这种位移过程的控制程序实际上是把程序 No. 5-7、No. 5-9 或 No. 5-8、No. 5-10 合并组成。合成后的程序见 No. 5-15、No. 5-16。

```
No. 5-15
LDX # $00
LDY # $00
LEFT STX $2005
STY $2005
JSR YANSHI
INX
BNE LEFT
DEX
RIGHT STX $2005
STY $2005
JSR YANSHI
DEX
BNE RIGHT
BEQ LEFT
5 REM No. 2-16
10 A=0
20 POKE 8197,A
30 POKE 8197,0
40 IF B=0 THEN A=A+1;IF A=256 THEN B=1
50 IF B=1 THEN A=A-1;IF A=0 THEN B=0
```


70 GOTO 20

在用汇编语言编写的连续位移程序中的延时程序,可能因延时时间短,使位移过程不易观察。No. 5-17是一个延时时间任意可调的通用程序。使用它可以像使用 PAUSE 语句一样方便。

LDA # \$a

LDA # \$b

JSR nn

RTS

程序中 a、b 的延时参数, a 为高位, a 值的一个单位相当于 PAUSE 256, 故一般取 a=0; b 为延时低位, 其取值与延时的对应关系与 PAUSE 语句相同; “nn”

为使画面的位移过程更易分辨直观,我们可以在执行位移程序前,把背景00、10两页都写入内容。常用的写入方法是在程序中加一条打印语句,向00页写入字符或图形;对于背景10页的写入有两种方法:一种是直接把字符数据写入PPU;第二种方法是在BG GRAPHIC画面绘制图案或字符(因为BG GRAPHIC画面使用背景10页面)。

在 10 页面最简单的绘图方法就是使用 BG GRAPHIC 画面键控绘图。操作过程是:BS. 2A 版本键入 SYSTEM 命令,返回选项画面,选择 2;若是 V3. 10 版本则键入 BGTOOL,都可进入 BG GRAPHIC 画面,键控绘图完毕,按 ESC、STOP 键返回 F BASIC,在程序中加入一行程序

这时再运行位移程序,就可清晰地观察到画面的连续位移过程。

~~~~~

将于 94 年 4 月 8 日~5 月 7 日举办微机二级(芯片级)维修班。由中国人民大学具有丰富实践经验的教师、工程技术人员任教。以典型疑难故障为例,通过现场排除故障演示,分析微机工作原理使学员掌握维修技术。

讲授内容包括:80286 主板(含 8088/80386/80486),软硬盘驱动器、适配卡,打印机(LQ1600),主机电源,显示器(CGA/EGA/VGA/SVGA),UPS。

学费:420 元 资料费:约 160 元

上课、食宿地点:中国人民大学

联系地址、联系人:北京中国人民大学信息中心 胡野红 胡智敏

邮码:100872

电话:2555431 转 3305 或 2563399 转 3305

# Microsoft Windows 3.1应用基础——基本操作(上)

中国科学院软件研究所 徐鹏力

上期介绍了窗口系统的基本概念、Microsoft Windows 的发展简史及 Windows 3.1新的特点与功能。下面将较详细地介绍 Windows 3.1的操作与使用技巧。

## 一 Windows 3.1的安装与运行

1. Windows 3.1的安装是由其装载程序(Setup)引导执行的。用户在 MS-DOS 状态下只需确认商品 Windows 3.1软盘的正确驱动器(A 或 B)位置并键入 Setup 命令、回车,就进入了安装过程。Setup 程序提供两种安装 Windows 的方式——常规安装和特殊安装。安装过程开始,用户必须选择一种安装方式。

•常规安装 由装载程序 Setup 自动检测计算机系统的配置情况并使用标准设置装载 Windows 程序。这种方式无需用户了解计算机系统过多的配置信息,适于大多数计算机用户或 Windows 初学者。

•特殊安装 这种安装方式需要用户有较多的计算机专业知识,在安装过程中由用户输入有关硬件、软件及其所需配置的信息。这种方式能优化运行 Windows 的系统环境,适用于计算机专业人员或已有 Windows 基础者。

无论是常规还是特殊安装方式,都需要计算机系统具备运行 Windows 3.1的基本条件(见上期“Windows 3.1的运行环境”)。在安装过程中,如遇到故障, F1(Help)信息将指导用户如何正确地完成安装。

2. Windows 3.1的操作模式 Windows 3.1根据系统硬件配置和使用需求的不同可有两种操作模式——标准模式和386增强模式。

若计算机具有至少1MB内存且使用 Intel80286微处理器,Windows 会自动地使用标准模式;若具有至少2MB内存且使用80386微处理器或更高档次,Windows 就会自动地运行在386增强模式下。运行标准模式时不具备多任务、虚拟存储及运行窗口式非 Windows 应用程序功能。

3. 运行 顺利安装完毕,启动 Windows 3.1只需在 MS-DOS 提示符下键入 Win,回车即可。Windows 会根据系统的配置自动选择一种操作模式运行。用户亦可键入 Win /s 使 Windows 从386增强模式运行状态改为标准模式。启动伊始,屏幕显示 Windows 3.1蓝色徽标,继而 Windows 的缺省外壳程序——Program Manager(程序管理器)。用户此时所见屏幕图案将由窗口及一些生动的肖像构成。这就是 Windows 3.1的缺省外观。

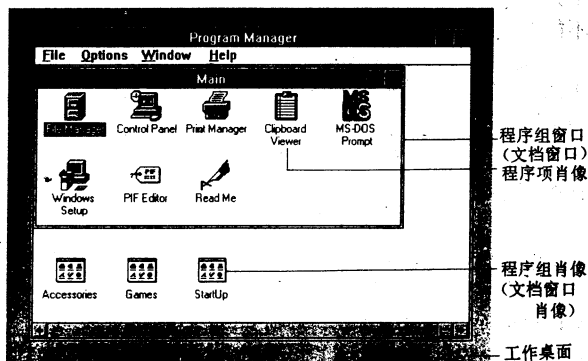


图1 启动 Windows 3.1

## 二 窗口基本组成

窗口与肖像是 Windows 图形界面最显著的外观特征,在 Windows 上所做的工作都将围绕图形窗口进行,为此,必须了解窗口的基本组成。

1. 窗口组成 尽管 Windows 有各种不同的窗口,但是绝大多数窗口都是由一些相同的元素组成(图2)。

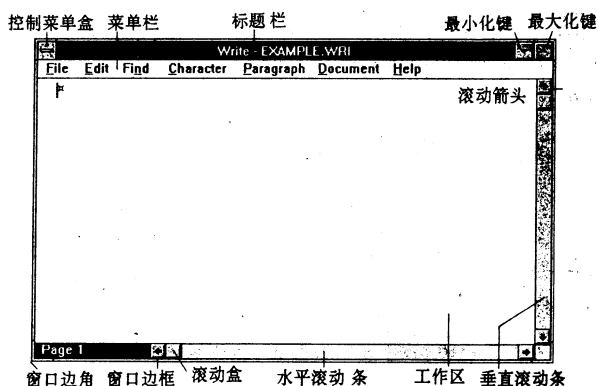


图2 窗口基本组成

•控制菜单盒 每个窗口左上角带有“一”的方框。单击(尽管用户可以键盘方式操作,但是使用鼠标器是 Windows 鼓励的操作方式,以下操作均以鼠标器为例)该方框可以打开控制菜单,其中包括对本窗口起作用的一些命令,双击则可关闭本窗口。

•标题栏 位于每个窗口的顶部边缘,显示本窗口的名字。拖动该栏可使本窗口在工作桌面上任意移动。

•菜单栏 位于标题栏下方,列出本窗口的可用菜

单。

•最大化和最小化键 位于窗口右上角。前者显示为▲,可使本窗口变为整屏大;后者显示为▼,使窗口变为小肖像。窗口最大化后,右上角的恢复键(◀▶)可使最大化的窗口恢复为原大小。

•边框 包括窗口边框和边角。拖动一条边框可在一个方向改变窗口的大小;拖动一个边角可同时在这两个方向上改变窗口大小。窗口最大化时边框看不见,也不能对其进行操作。

•滚动条 包括垂直滚动条和水平滚动条。当显示内容超过窗口大小时,利用滚动条上下或左右滚动可看到当前窗口无法显示内容。

•工作区 是窗口中间的部分。打开的子窗口或文件等都显示在这工作区内。

•鼠标光标 随鼠标器移动而移动,用来标识和进行选择。

2. 窗口类型 用户在实际操作过程中常会看到 Windows 3.1 的两种窗口——应用程序窗口和文档窗口。

•应用程序窗口 这种窗口包含正在运行的应用程序,应用程序的名字、相关文档及菜单栏都显示在窗口的顶端。

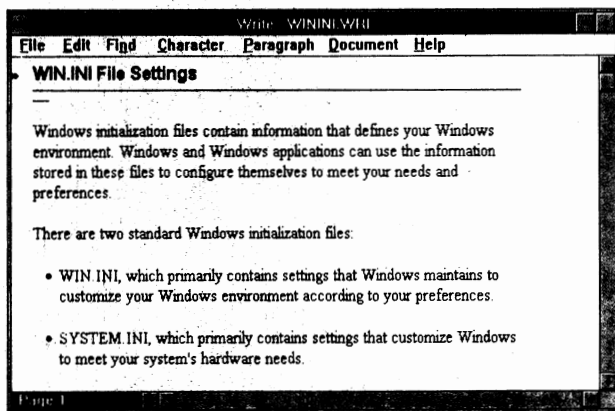


图3 应用程序窗口

•文档窗口 这种窗口都出现在应用程序窗口中,没有自己的菜单栏,而是与应用程序窗口共享菜单栏,但它有标题栏。一个应用程序窗口可以同时打开多个文档窗口。最大化一个文档窗口时,它并不占满整个屏幕而是占满整个应用程序窗口工作区,并与应用程序窗口共享一个标题栏。文档窗口不能和应用程序窗口一样在工作桌面上任意移动,它只能在应用程序窗口工作区内移动。

图例为同时打开的三个文档窗口(Main、Games、Accessories)与一个应用程序窗口(Program Manager),请注意它们的区别。

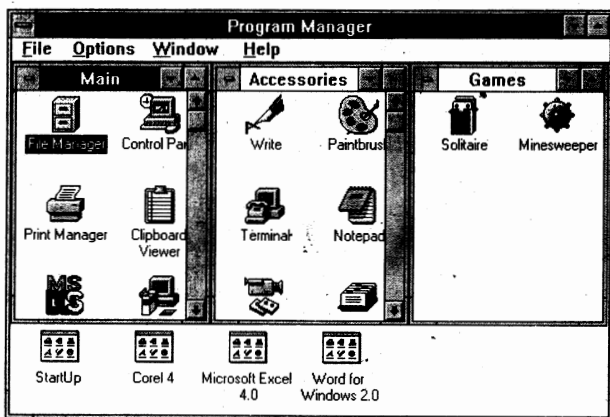


图4 文档窗口与应用程序窗口

3. 肖像 肖像是 Windows 图形操作环境中各种元素的一种图形表示方式。它可以代表正在运行的应用程序、文档窗口以及可启动的应用程序。Windows 3.1 提供了三种肖像。

•应用程序肖像 对于一个正在运行的应用程序,如果选择了控制菜单盒中的 Minimize 命令或直接选中该窗口上的▼键,该应用程序就最小化为一个肖像显示在屏幕左下角而继续运行。利用鼠标器在应用程序肖像上连击两次,它又恢复为原来的应用程序窗口。

•文档肖像 如图1所示,Accessories、Games、StartUp 等都是文档肖像(有时也称文档窗口肖像)。当我们在打开的文档窗口 Main 的控制菜单盒“一”上连击两次,文档肖像 Main 将显示在应用程序窗口的左下角。当我们在其肖像上再次连击两下,它就又恢复为原来打开的 Main 文档窗口。

•程序项肖像 Windows 中又称文档窗口肖像为程序组肖像。例如图1中就有 Accessories、Games、StartUp 等程序组肖像,而 Main 则被称为打开的程序组窗口,对应于程序组中的每项称为程序项,每一个程序项都代表着不同的应用程序,在程序项肖像上连击两次将启动该肖像所代表的应用程序。例如在 Main 组中,在 File Manager 程序项肖像上双击将启动文件管理应用程序。

### 三 菜单的操作

菜单是 Windows 图形界面提供文字信息的重要工具。它是一种命令列表,从菜单中选择命令是 Windows 程序中完成某项工作最常用的方法。每个窗口左上角都有一控制菜单盒(见图2),盒中的菜单命令是 Windows 每个窗口几乎都公共的。每个应用程序窗口都有自己的菜单栏,栏上每个菜单名都可以打开一个菜单,它们包含了本应用程序窗口所能执行的命令。例如,图3中 Write - WININI.WRI 应用程序的 File 菜单包含了用于打开及保存文档文件的命令,而 Edit 菜单则含有改变文档内容的命令。

#### 1. 选取和撤消菜单

选取:单击菜单栏上的菜单名,窗口将弹出一下拉式菜单。

撤消:单击菜单名或下拉式菜单外的任何地方。

## 2. 选取菜单命令

对于被选取的菜单所弹出的菜单命令选择,只需鼠标单击所需的命令即可。对于菜单命令,Windows 系统有如下约定:

- 变灰的命令:表明本命令当前状态下是无法起作用。
- 命令前有√:表明本命令当前状态下已在起作用。
- 命令后有…:表明本命令是一个未完成的命令,选取它将带出一个对话框,以使 Windows 能获取更多的信息。
- 命令后有▶:表明选择本命令后将列出第二辅助式菜单(又称级联式辅助菜单)。
- 命令后有键盘快捷键:表明使用该快捷键同样可以执行本命令。

当然,在 Windows 系统中还有变化的菜单命令,用户可以修改的菜单命令等方法,这些以后讨论。

## 3. 使用控制菜单

应用程序窗口、文档窗口、应用程序肖像、程序组肖像及一些对话框都有控制菜单。

打开:单击窗口的控制菜单盒或屏幕上的肖像。

以下所列的控制菜单命令一般是 Windows 控制菜单盒中所公共的,但并非每个窗口控制菜单盒中都含有下述的命令。

- Restore(恢复):恢复被最大或最小化的窗口为原来大小。
- Move(移动):使用键盘移动窗口或肖像。
- Size(大小):使用键盘改变窗口大小。
- Maximize(最大化):扩展窗口为最大。
- Minimize(最小化):收缩窗口为肖像。
- Close(关闭):关闭窗口。
- Switch to…(切换):启动任务列表,完成运行的应用程序间的转换或在工作桌面上重新安排窗口的肖像。
- Next(转移):仅对文档窗口有效,完成在打开的文档窗口和肖像间的切换。

## 四 对话框操作

对话框也是 Windows 系统中提供文字信息的重要工具,它是一种特殊的小窗口,列出各种选择项要求用户提供信息。几乎所有的对话框都具有如下的组成部分:

- 一个或多个用于输入信息的地方或可选取的任选项。
- 一个或多个“命令按钮”。

下面以图5、图6为例介绍常见的对话框组成形式:

1. 命令按钮 图5右边五个标有命令的矩形按钮,单击执行相应的操作命令。

2. 列表框 Name 即为一列表框,它含有一系列可选项。有时单击列表框所带的下箭头将弹出一个下拉式列表,用户进行选择只需在某一项目上单击鼠标(图

6中 File Name 亦为一列表框)。

3. 选项按钮 Wallpaper 框内的 Center 和 Tile 即为一组相互排斥的选项。黑色小圆点在每组中同时只能选取一个选项按钮,代表该选项当前有效。

4. 文本框 图6中 File Name 下长矩形框即为文本框,用户可从键盘输入文本(字母、数字和符号)。

5. 组合框 列表框和文本框的组合,File Name 即为一组合框。组合框允许从显示的列表框中选取信息,也允许在文本框内直接输入文本。

6. 选择框 单词或短语前的小方框。图5中 Wrap Title 即是。方格中出现×时表明该选项被选中,反之该选项则被关闭。在一个对话框内,可有几个选择框同时被选中。

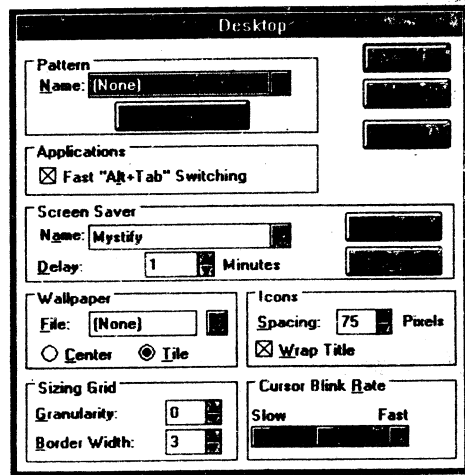


图5 对话框示例一

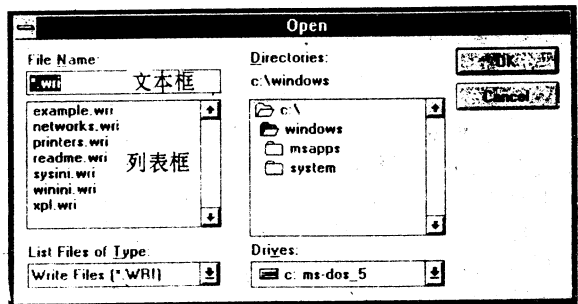


图6 对话框示例二

## 二 PC Tools 文件服务功能

王路敬

### 1. 文件服务功能主菜单

当启动 PC Tools 后,屏幕上显示下列信息:

WELCOME!

PC Tools Deluxe R4. 21

(C) Copyright 1986, 1987, 1988 Central Point Software, Inc.

Unauthorized duplication prohibited.

Press any key for File Functions

OR

F3=go directly to Disk and Special Function

F10=change drive/path from A:\

Press ESC to Exit

上述英文提示信息的中文含义是:

欢迎!

PC 工具高级版本4. 21

拷贝1986,1987,1988 Central Point Software 公

司

禁止非授权复制

按任意键进入文件功能

或

F3=直接进入磁盘和特殊功能

F10=从 A:\改变驱动器/路径

按 ESC 退出

此时,用户按任意键,即可选择文件服务功能,则  
屏幕显示文件功能主菜单如下:

PC Tools Deluxe R4. 21

Vol Label=None

File Functions

Scroll Lock OFF

Path=A:\\*.\*

| Name     | Ext | Size   | Attr  | Date     | Name | Ext | Size | Attr | Date |
|----------|-----|--------|-------|----------|------|-----|------|------|------|
| PCTOOLS  | EXE | 170996 | ... A | 11/20/87 |      |     |      |      |      |
| PCBACKUP | EXE | 47032  | ... A | 11/13/87 |      |     |      |      |      |
| PCRESTOR | EXE | 52872  | ... A | 11/18/87 |      |     |      |      |      |
| PCFORMAT | COM | 12257  | ... A | 11/17/87 |      |     |      |      |      |
| PC-CACHE | COM | 6872   | ... A | 11/19/87 |      |     |      |      |      |
| PC SETUP | COM | 10048  | ... A | 11/14/87 |      |     |      |      |      |
| COMPRESS | EXE | 36860  | ... A | 11/20/87 |      |     |      |      |      |
| MIRROR   | COM | 12119  | ... A | 11/19/87 |      |     |      |      |      |
| REBUILD  | COM | 13057  | ... A | 11/19/87 |      |     |      |      |      |

|                  |   |               |                     |   |               |
|------------------|---|---------------|---------------------|---|---------------|
| 9 files LISTed   | = | 337737 bytes. | 9 files in sub-dir  | = | 337737 bytes. |
| 0 files SELECTed | = | 0 bytes.      | Available on volume | = | 22528 bytes.  |

Copy Move cOmp Rename Delete Ver view/Edit Attrib Wordp Print List

Sort Help =SELECT F1=UNselect F2=alt dir 1st F3=other menu Esc=exit PC Tools

F8=directory LIST argument F9=file SELECTION argument F10=chg drive/path

功能主菜单中有关信息的含义说明如下:

Vol Label:磁盘卷标,如果没有卷标,则显示为

None。

File Function:当前所用功能为文件功能

Scroll Lock OFF:上卷锁开关。按 Scroll 键可使其  
开(ON)或关(OFF)。

Path:当前路径。现指 A 盘根目录。

Name:文件名

Ext:文件扩展名

Size:文件大小,单位为字节。

Attr:文件属性。

S 表示系统文件

H 表示隐含文件

R 表示只读文件

A 表示归档

Date: 文件建立日期。

文件功能的使用方法简介如下:

进行文件功能的操作, 首先选择将要操作的文件, 选择的方法有两种:

(1) 若只选择一个或多个文件且文件名无规律, 可通过移动光标来完成。用于移动光标的键有:

↑ ↓ → ← — 可使光标上下左右移动

Home (End) — 将光标移到目录的第一(最后)一个文件上

PgUp (PgDn) — 将光标向上(下)移动六行

当光标被移到要选择的文件上时, 按回车键, 即确认选择。

(2) 若选择多个文件, 且文件名或其扩展名有一定规律, 则可根据菜单提示按 F9 功能键选择文件, 按下 F9 后, 屏幕将显示一个空白的文件名和扩展名等待用户输入, 这时可用通配符“?”或“\*”进行输入, 该功能将根据用户的输入内容选择文件。

如用户输入以下内容:

Name = [ \* ]

Ext = [ EXE ]

则表示选择所有以 .EXE 为扩展名的文件。其中, 方括号中的内容需要用户输入。

选择后的文件可取消, 取消个别文件可直接将光标移到该文件上再按回车, 则黄色背景消失, 顺序号也同时消失, 其它顺序号将重新排定。若要去掉全部文件, 按 F1 (消去选择) 功能键, 则全部选择文件被取消。

除 F1、F9 回车键外, 其他功能键的功能是:

F2: 当按下 F2 后, 屏幕上文件目录的显示将发生变化, 不仅显示出文件名、扩展名、长度、文件建立时间及属性, 还将显示文件所占的簇数。

F3: 按下 F3 键则将显示磁盘服务功能的菜单。

F8: 按下 F8 后, 屏幕会显示出一个空的文件名和一个空的扩展名让用户填写, 用户可利用通配符“\*”, “?”填写出有关信息, 填写后, 按回车键则在目录区显示有共同特征的文件。

例: Name = [ \* ]

Ext = [ COM ]

回车后目录区显示出当前目录区以 .COM 为扩展名的所有文件目录。

若期望改变当前路径, 可按 F10, 屏幕显示一窗口:

```
Enter NEW drive letter below. Press
" "for no change, "Esc" to return.
NEW Drive ID - [C]
Valid letters are A thru E.
```

窗口里英文的含义是, 输入一个新的驱动器, 若不改变就按回车键, 按 Esc 返回。

文件服务功能有以下一些内容供选择:

(1) Copy (文件拷贝)

将文件按顺序从一个驱动器上的软盘拷贝到另一驱动器上。该功能的特点:

① 可以按一定顺序拷贝文件。使用回车键可以选择拷贝顺序, 被复制的文件的顺序会按后选择的顺序安排。

② 使用 F9 可以同时拷贝多个文件名有规律的文件或当前路径中的全部文件。

③ 按 Esc 键随时可中止拷贝功能的执行, 返回到文件功能主菜单。

[例1] 将 A 盘上的 .EXE 文件拷贝到 B 盘

操作步骤:

① 按 F9 功能键, 选择 A 盘 .EXE 文件。使用箭头移动光标, 选择所需目录, 按回车键承认选择。

② 按“C”, 屏幕弹出窗口:

```
Enter TARGET Drive ID - [A]
Valid letters are A thru E.
Press ESC to return
```

窗口中信息的中文含义是:

输入目标驱动器标识 - [A]

有效字母为 A-E

按 Esc 返回。

③ 输入有效驱动器号“B”后回车, 屏幕显示:

```
Path = A:\
To Path = B:\
xxxxxxx. xxx is being copied
Press ESC to Exit
```

④ 在运行过程中, 可能出现下列情况

情况一: 目标盘没有足够的空间

此时屏幕显示:

```
Not enough free space on the
To drive to hold all the files
selected
```

Press any key to return

在这种提示下, 按任意键返回, 减少一些文件后再重新拷贝。

情况二: 目标盘存有同名文件

此时屏幕显示:

```
path = A:\
To path = B:\
xxxxxxx. xxx is being copied
File already exists
Press "R" to replace all files that exist
"W" to replace this file only
"S" to skip all files that exist
"T" to skip this file only
or Esc to terminate operation
```



在这种情况下可进行如下处理:

- 按“R”键,则替换所有现存文件
- 按“W”键,只替换当前现存文件
- 按“S”键,跳过所有现存文件
- 按“T”键,只跳过当前现存文件
- 按“Esc”键,中断文件拷贝。

[例2]将C盘根目录下的PCTOOLS.EXE文件拷贝到C盘的TOOLS子目录下。操作步骤:

- ①在C盘根目录下选择PCTOOLS.EXE文件
- ②按“C”键,屏幕弹出菜单让用户输入目标驱动器标识符
- ③按“F10”键后回车,屏幕显示目录结构,然后通过↑↓→←移动光标键将光标移到TOOLS子目录下并回车,则PCTOOLS.EXE会被拷贝到指定的TOOLS子目录下。

作为文件拷贝功能与DOS中COPY命令相比,在顺序拷贝上更加方便,并且具有拷贝隐含文件的功能,但由于其拷贝路径是通过改变屏幕的方式进行,因而使用起来速度比较慢。

## 谈谈COMMAND.COM

王伟廷 锦州凌河区文昌里4号(121001)

我们日常所用的磁盘操作系统一般为PC-DOS或MS-DOS。以MS-DOS为例,它包含有以下系统文件:

IO.SYS  
MSDOS.SYS  
COMMAND.COM

其中前两个的属性是隐含的、系统的,内部命令如COPY、DIR及DEL等对它们都不起作用。而COMMAND.COM却是可见的、可删除的、可拷贝的,甚至是可更改的。实际上,COMMAND.COM不仅是一个系统文件,它还是一个外部命令。

1. 作为一个系统文件,系统启动时,COMMAND.COM分为常驻部分和暂驻部分。常驻部分和IO.SYS及MSDOS.SYS一样一直存放在内存中,暂驻部分(包括执行内部命令的程序)当运行一些应用软件时为节省内存而被覆盖。当退出应用软件的运行时,还要把暂驻部分加载到内存中,这就要求COMMAND.COM存放在由SET COMSPEC=指定的路径中(默认系统启动时的驱动器)。否则将会提示操作者插入含有COMMAND.COM文件的软盘或死机。例如,在没有硬盘的情况下,如果您运行文字编辑软件CCED或工具软件PC Shell,将会明显遇到这类问题。

2. 作为一个外部命令,在DOS状态下执行没什么

太大意义。但如果在dBASE III Plus(或FoxBASE)及BASIC状态中,有时因为某些需要欲暂时回到DOS状态,如果退出dBASE III Plus或BASIC会很麻烦,这就必须运行COMMAND.COM。在dBASE III Plus中可打入RUN COMMAND回车;在BASIC中打入SHELL“COMMAND”回车,都可进入DOS状态。要返回到dBASE III Plus或BASIC状态,只需在DOS提示符下打入EXIT回车即可。

3. 在编辑批处理文件时,如果一个批处理文件中间还要启动另一批处理文件(即批处理文件的嵌套),那么就必須用如下格式:

```
COMMAND /C filename2
```

其中filename2为被调用的批处理文件的文件名。否则,以后的命令将不被执行。

4. COMMAND.COM可以通过DEBUG或PC Tools修改。例如,你修改字符BAT为BBB,那么扩展名为.BBB的文件将作为批处理文件执行。你修改字符DIR为HHH,那么HHH将作为内部命令用于显示目录。如果知道这点,你可以使用自己的COMMAND.COM,以达到对文件或系统保密的目的。

(上接56页)

```
8,fp))
{
for(i=0;i<num;i++)
{
file[i]=file[i]^password[i];
printf("%c",file[i]); }
if(num<8)break;
}
fclose(fp);
return 0;
}
```

此程序同时提供了一种破译WPS密码的方法,对此程序略一改动便会实现密码读出功能。对于WPS能编辑的“非文书文件”阅读器的制做更简单,只须直接读出数据即可,本文不再赘述。

(上接51页)同时,备份命令还会在硬盘上(Backup所在目录)建立一个备份目录,它包括以下信息:

- \* 备份盘的目录结构
- \* 选定的目录或文件的名、大小和属性
- \* 备份文件的总数
- \* 备份的总长度
- \* 所用配置文件的文件名
- \* 备份日期

备份命令并给它一个专用文件名,其名前部分与备份文件生成的文件名相同,后缀为FUL、INC或DIF,其含义为:

FUL 备份类型为完全备份  
INC 备份类型为增量备份  
DIF 备份类型为差异备份

# 扫描仪输入技术及应用

方卫宁 何晓波(重庆大学) 邹华(北京邮电学院)

## 一 前言

我们在信息社会中所获得的信息有70~80%来源于视觉,计算机作为信息社会的主角,已经能看、能听、能说和能推理。计算机图象处理在许多领域起了关键的作用,但计算机处理图象信息,最初都是借助图象输入设备将自然图象翻译或变换成为能被计算机接受和使用的数字图像,图象扫描仪正是实现这一过程必不可少的工具。近几年来随着科技的迅速发展,利用计算机处理资料的电子资料系统日益受到重视,管理电脑化已经成为未来世界的必然趋势。

图象扫描仪(Image Scanner)自从1984年问世以来,至今已成了办公自动化中不可缺少的一项设备,它的功能特性在这数年之中,从只能扫描黑白图象到具有灰度识别能力,再发展到能扫描彩色图象,其间技术的更新、产品的发展,可以说是日新月异。扫描仪的应

用范围已经从桌面排版、光学字符识别覆盖到计算机辅助设计、数据库管理等方面。由于人们总在不断思考着如何将它的功能发挥到淋漓尽致,这将使得扫描仪输入技术在今后的社会中扮演着更为重要的角色。

## 二 扫描仪结构剖析

扫描仪是一项结合光学、电子、机械控制等尖端科技的产品,其中包含了光学原理、电荷耦合元件(CCD)、模拟/数字转换电路、数字信号处理(DSP)、图象压缩与传送等技术。(图1)

扫描仪依其尺寸大小及使用目的可分为:

1. 手持型扫描仪:以个人及商业用途为主。
2. 桌面型扫描仪:主要用于桌面排版系统及文档自动化管理系统。
3. 大型扫描仪:以工程图纸管理系统、计算机辅助设计及地理咨询系统上的应用最为普遍。

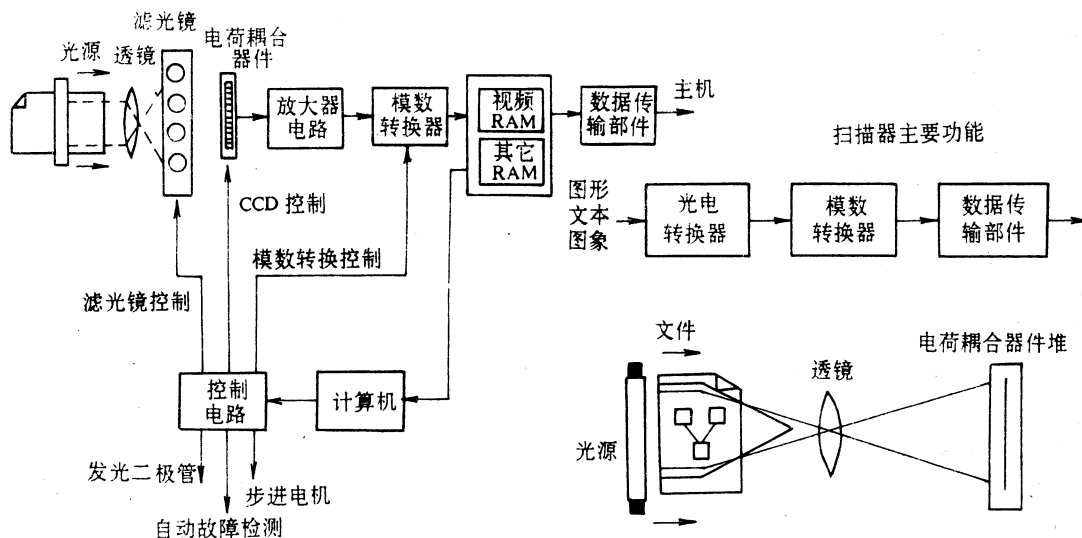


图1 图象扫描解剖图

扫描仪的主要内部结构如下:

### 1. 光学原理

一般来讲扫描仪可分为反射式以及穿透式光源,所谓反射式即是光源投射在被扫描文件,经透镜反射聚集到CCD,从而产生一连串的电子信号;而穿透式光源即是所谓的幻灯片式扫描仪(Slide Scanner),其

图2 图象扫描光学原理

光源是穿透过被扫描的文件,最重要的是如何精确对焦,使经由透镜反射的光能正确成象于CCD。(图2)

### 2. 电荷耦合元件(CCD)

这部分可以说是扫描仪的核心,在CCD上直线排列的光或元素可因入射光强弱,以电压的形式反应出来;它主要的功能在于将由透镜聚集的光转换成电子

信号,使得扫描仪能“看”见这些文件。

### 3. 模拟/数字转换电路

由 CCD 产生的信号通常是非常微弱的,需要用信号放大器(Amplifier)加以放大,但由于这是模拟信号,所以必须用模拟/数字转换电路转换成数字信号,以便传输给计算机进行处理。

### 4. 图象传输处理器

为了让扫描入的图象能在各种计算机平台上处理,这部分必须提供与计算机的接口功能,使用户可以在 IBM-PC、Macintosh、Sun Workstation 等主机上处理或使用扫描进来的图象资料。

扫描仪与计算机之间通常有两种接口:一是并行接口,由邮箱通道、存储器直接访问通道(DMA)组成。邮箱通道是双向通道,它可以用来传输命令、信息和数据;DMA 通道是单向通道,它只能用来传送从扫描仪到计算机的扫描数据。并行接口传输效率高,目前被广泛采用。二是串行接口,一般使用的是 RS-232C 接口。

## 三 扫描仪特性分析

扫描仪的性能主要包括以下几个方面:

1. 扫描区域:大致而言,目前市面上的扫描仪扫描图象的宽度因扫描仪的大小而受限制,扫描宽度一般可从105mm到A0号图纸幅面宽,具体视扫描仪的型号而定。从纵向而言,大部分的图象扫描仪(工程型扫描仪)可以无限延长,此外,扫描仪所能扫描的材料可包括证券用纸、塑料薄板以及亚麻制品等等。

2. 分辨率:分辨率的大小一般以“点/每英寸”(dpi)为单位,大部分的图象扫描仪能够以75dpi-1000dpi范围内的分辨率来扫描。有些扫描仪甚至允许用户选择他们想要的分辨率来扫描。虽然有少部分图象扫描仪的分辨率超过了上述范围,然而就一般而言,这样高的分辨率目前并不需要。扫描仪分辨率的选择,可依据图象原稿的清晰情况,以及文件的用途来决定。

例如要执行“图象——矢量”(Raster to Vector)的转换,则较高的分辨率就比较有利,相对地,若只是要将扫描得到的图象存储起来,或者用图象硬拷贝机(Raster hard copy device)印制出来,则较低分辨率的图象扫描仪就足够了。

3. 灰度级:有些扫描仪能在黑与白之间分出许多不同层次的灰度,我们称之为灰度级。一般图象扫描仪能够分出的灰度级在8到256之间。理论上来说,不可能同时要求高的灰度级以及高的分辨率,如果要求高的灰度级,则分辨率便会相对降低,反之亦然。

以工程上的应用而言,灰度级通常用于扫描图纸中细微的部分以及不明显的线条,而对技术性印刷而言,在扫描“半色调”(half tone)或有灰度的图象方面,灰度级的扫描更能显现其重要性。

另外,一些新型的扫描仪已经具有了处理彩色图象的能力,在某些工程的应用上,这种彩色扫描仪尤其有用。例如将 CAD 中建立的模型,放到各种不同的背

景中,来了解这种实体在不同的环境下所显现的面貌,而这些背景资料正可以用彩色扫描仪来完成。目前市面上彩色扫描仪的彩色模式已达到24位,可以处理大约1670万种颜色。

4. 图象处理:图象处理的功能可以使扫描的结果更为清晰。例如:边缘加强功能,能够捕捉原稿上边缘有缺陷的部分,而噪声滤波功能则可以去掉原稿上的污点及毛刺。值得一提的是,由于这些图象处理的功能是与扫描过程同时进行的,因此并不会将扫描速度降低多少。

5. 精确度及可重复性:图象扫描仪精确度的计算大致与笔式绘图机的计算方式一样,分为横向扫描精度和纵向扫描精度;可重复性则是测试当扫描仪多次扫描同一文件时,所产生图象是否完全相同的一种能力,图象处理功能以及高精度度能够使扫描仪产生更佳的图象,而这种高品质的图象资料不但可以使“图象——矢量”转换得到更好的结果,甚至还可以节省10%~20%的存储空间。

6. 编辑与速度:有些扫描仪允许用户选择文件中的某些区域来扫描,因此可凭借扫描区域的缩小来减少扫描时间。

扫描速度的测量单位以英寸/每秒(ips)来表示,以200dpi分辨率的扫描而言,扫描速度通常在0.25~2ips之间,一般来说,扫描的速度与分辨率以及灰度级有关。虽然扫描一张文件的时间,某一台扫描仪可能只比另一台快上几秒钟,但如果所要扫描的文件高达成百上千张时,那么所累积的时间就很可观了。

## 四 扫描仪的应用

扫描仪本身只是计算机的一种输入装置,但目前随着应用软件的蓬勃发展,促使扫描仪发挥出越来越大的作用,下面我们就来谈谈扫描仪的各种应用。

### 1. 桌面电子排版

由于表格、图象、照片、文字混编的需要,使得桌面排版成为扫描仪最大的使用群。扫描进来的图象可通过如 Corel Draw、Illustrator、Arts & Letter 等绘图软件或随机附带的软件做一些旋转、轮廓修正等编辑后,再配合桌面排版软件如(Page Maker、Ventura、Publisher、Frame Maker),将图象与文字混合编辑,制作出一篇多姿多彩、图文并茂的设计。另外彩色图象扫描仪的推出,也使得传统的彩色分色印刷转移到 Macintosh 等计算机上,可以让美术师、广告及包装设计使用 QuarkxPress 或 Page Maker 来设计初稿,再送往专业彩色印刷厂家制作出最后的完稿。

### 2. 光学字符识别 (Optical Character Recognition)

光学字符识别 OCR 软件的开发使扫描仪不仅能扫描文字稿,同时还具有了智能识别能力,它能将扫描识别过的字符以 ASCII 码或其它文字处理软件的格式来存储,这样不仅节省了原先以图象格式存储的空间,还减少了重新键入的麻烦,加快了文件处理的速

度。现在已有些厂商将 OCR 功能加入到扫描仪中,使它成为一种智能型的仪器,并且进一步加快了识别的速度。

### 3. 计算机辅助设计 CAD

扫描仪应用在 CAD 领域一直被认为是一项有相当潜力的市场。一般桌面型扫描仪通常只能扫描 A4 (8.5"×11") 尺寸的幅面,由于工程图、建筑图、地图的幅面尺寸较大,因此大型扫描仪应运而生,解决了这类图纸的输入问题。目前的应用是将原有的工程图纸扫描输入进来,经过矢量化软件或其它人工智能系统处理转换成 CAD 系统所能接受的数据格式,在 CAD 系统中对图纸进行编辑、修改和存储,紧接其后的应用便是整个工程图文档系统的建立与管理、地理咨询系统的应用等等。

### 4. 计算机传真系统

计算机传真系统与扫描仪的配合使用,可以使计算机成为图象资料传输的工作站,再配合其它图象编辑软件,将比一般传统的传真机具有更大的灵活性、经济性和应用空间。

### 5. 图象数据库

传统的数据库只能存储文字资料,扫描仪的推出,使得这些数据库系统加入了图象文件的存储,使资料更具有“可看性”,如果将文档扫描输入光盘,同时引入索引、检索等功能,则可形成一综合的图象数据库。

## 五 扫描仪的选择

综观扫描仪市场,真可谓百家争鸣、各有千秋,究竟该如何选择最适用的机种呢?可以从以下几点来考虑:

#### 1. 硬件的特性:

一般谈到的扫描仪大多提到分辨率(300/600/800...dpi)与图象的灰度级(gray level、16/64/256级)或每个象素对应的字节位数(bits/pixel、4/6/8/12/24bits),有的扫描仪提供了内部图象灰度修正功能,以便得到更高品质的图象资料。适当分辨率的选择,可以避免一味追求高分率而造成设备采购上的浪费。一般

用于工程处理的扫描仪300—600dpi 便足够了,地图的绘制应选择400—1000dpi 间的扫描仪,1000dpi 以上则属于某些专业领域如印制电路板布线的应用范围。

#### 2. 软件的配置

配置的软件是否能提供各种图象格式(如:TIF、PCX、GIF、IMG...等)以及与其它软件或不同系统的接口;是否提供了图象重整(Image Resoration)、彩色平衡与修正(Color balance/correction)等功能,以便能展现更清晰、传神的图象资料。

在选购用于工程目的的大型扫描仪时,必须配置与之相适应的专用软件,这样才能充分发挥出扫描仪的效益。如选购的扫描仪用于工程图纸管理系统时,则可配制如图象查询软件(Raster Viewing)、矢量化转换软件、关系数据库管理系统软件等,这都可增强扫描仪系统的功能。

#### 3. 用户要求

究竟为何要来采购扫描仪?主要用于那一种用途?单位现有计算机上的 RAM 容量多大?是否有足够大的硬盘来存储图象资料?主机性能能否满足与扫描仪配套软件的环境要求?视频卡是否能逼真地显示彩色扫描等。这些都是必须考虑的因素。另外市场价格、产品性能、销售厂家的售后技术服务等也是用户在购买时应注意的方面。最后再在精打细算中选择一种最佳的机型。

## 六 结束语

扫描仪的发展至今已有很多年了,从无到有,从黑白到彩色,由于用户的广泛支持和不断需求,各制造厂商都在绞尽脑汁,精心设计,努力研究开发更精良、更高层次的产品,以满足广大用户的需求。可以预计在不久的将来,扫描仪必须向着更高分辨率、更快的扫描速度、更好的扫描品质方向发展。随着我国市场经济的进一步深入以及电子计算机技术的飞速进步,扫描仪输入技术的应用及普及将是势在必行,它在自动化管理系统和我们的社会生产中将会起到越来越重要的作用。

# 利用 PC Tools 7.0 实现 PC 机间的高速通信

江苏徐州空军后勤学院航材系(221000) 程启超

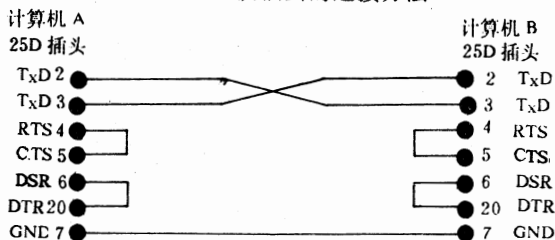
在空军航材管理信息系统的开发及应用过程中,存在着计算机间大量的数据传输工作,利用软磁盘进行拷贝比较麻烦且耗费大量的软盘。因而考虑到用 RS232 通信口进行文件传送。大家知道,PC Tools 7.0 中有一个重要的软件包:Desktop Manager,该程序包括无错误传送文件的通用 XMODEM 传送协议,可以

实现近域及远程计算机间的通信。对于快速文件传送,该软件包提供了 Desk Connect,你可以在未联网的计算机之间使用它来传送文件,一台计算机仅作准备,而另一台接管,选择来自其中一台计算机的驱动器上的文件,并在其中一个目录中传送文件。笔者经过近两年的实践,取得了较好的效果。

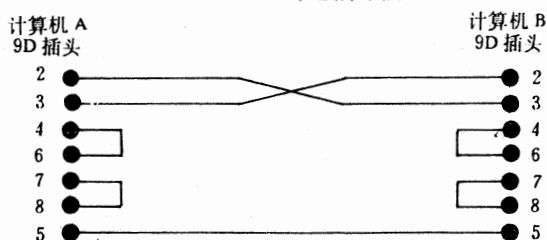
## 一 硬件连接方法

笔者采用三线连接的方法:

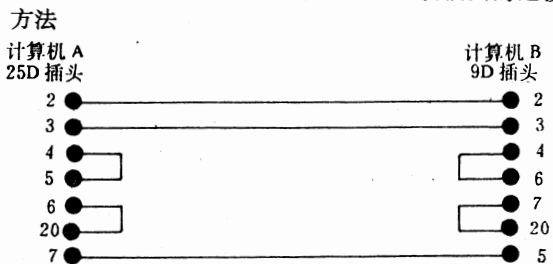
### 1. 两个 DB-25 芯阴插头的连接方法



### 2. 两个 DB-9 芯阴插头的连接方法



### 3. 一个 DB-25 芯和一个 DB-9 芯阴插头的连接方法



上述连接属硬接线的连接方法,不需要调制解调器(或称“空调制解调器”连线法),其通信线不能太长(一般在四米以内),否则会影响传输速率。

## 二 软件使用方法

1. 先在从机上拷入 PC Tools 7.0 软件包中的 DESKSRV.EXE 程序。如果是用接线直接连到 COM1,则只要送入下列命令:

DESKSRV

如果是接到 COM2,则必须告诉它:DESKSRV -C:2;如果是接到 COM3 或 COM4,则还必须告诉它 IRQ 和基本端口地址。在标准的 PC 兼容机上,用于 COM3 的参数是:DESKSRV -C:3 -I:4 -P:3E8;用于 COM4 的参数是:DESKSRV -C:4 -I:3 -P:2E8。

运行上述命令后,将在屏幕上显示 PCTools 7.0 的版本信息及 Comport:COM1,Baud:115200,Status:Waiting、Receiving、Sending 等字样。这台计算机在你退出程序之前将是台死机。

2. 在主机上装入 Desk Connect,送入下列命令:

DESKCON

除 COM1 之外的那些端口用的参数与 DESKSRV 用的相同。

接着,装入 PC Shell,选择 Special 菜单下的 C (DeskConnect) 命令。如果你的“空调制解调器”线缆连接正确了,并且正确地标识了各个端口,则将看到 1. 中的屏幕框图及内容。选择 OK,然后就会在你的驱动器条目上看到从机上所有附加的驱动器字母。

现在,你就可以管理从机上的文件,或以任一方向传送文件,从机上的那些驱动器就象在你的主机上的一样。例如,设主机上有 A、B、C 三个逻辑盘(软驱和硬驱),从机上有 A、B、C 三个逻辑盘,经过上述操作后,从机上的 A、B、C 三个逻辑盘将映射为主机上的 D、E、F 三个逻辑盘。这里,PC Shell 的所有命令和函数,除了从机上的盘片格式化之外,都可供你使用。

笔者应用上述方法成功地传送了大量的数据库中间过程文件,且波特率为 115200,传送速度相当高,从而实现了两台计算机间的最简单的高速通信。有兴趣的读者不妨一试。

## 使用 DOS 6.0 经验一则

陈乃强 李志 河北师范学院

使用以前的 DOS 版本(DOS 3.30、DOS 3.0 等)中的 Backup 命令生成的文件名为 Backup.001、Backup.002 等依此类推。

而使用 DOS 6.0 中的备份命令(Msbackup)生成的文件名却不像如上所述的那样。现举一例说明:我在 11 月 7 日使用了 Msbackup 命令从 D 盘向 B 盘备份文件,生成的文件名依次为 DD31107A.001、DD31107A.002 等,其文件名中的后缀 001 和 002,我们都清楚它指明的是本盘为几 # 盘,而前部分却有如下面所述的含义:

| 字符 | 含 义                                                                                                                                               |
|----|---------------------------------------------------------------------------------------------------------------------------------------------------|
| D  | DOS 6.0 规定的备份集的本集中第一个备份驱动器                                                                                                                        |
| D  | 本集中最后一个备份驱动器(如果只备份了一个驱动器,则此字母与第一个字母相同)                                                                                                            |
| 3  | 年的最后一位,这取决于计算机上的日期设置,本例中为 1993                                                                                                                    |
| 11 | 建立此备份的月份,它也取决于计算机上的日期设置                                                                                                                           |
| 07 | 建立此备份的日期,它也取决于计算机上的日期设置。                                                                                                                          |
| A  | 此备份在一列备份中的位置。若在同一天对同一个驱动器做了多个备份,Keep Old Backup Catalogs 项为 On,Backup 就会分配一个 A 到 Z 之间的字母来指明备份次序。若设置 Keep Old Backup Catalogs 项为 Off,则此字母轮流为 A 和 B |

(下转 47 页)

# 交流有效值—直流变换 IC

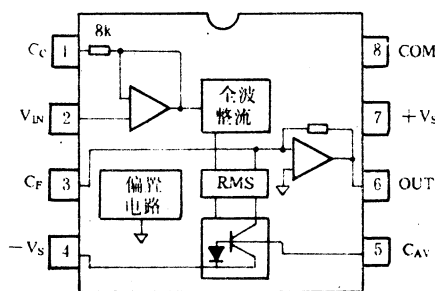
## AD736/AD737及应用

王俊省 李兰友

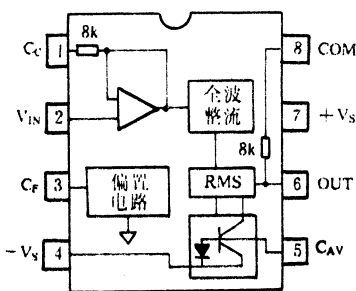
AD736/AD737是美国 AT 公司开发的低价格有效值——直流变换 IC。本文介绍其主要特性及使用方法。

### 一 引脚及参数

AD736、AD737引脚及内部结构方框图如图1所示,图1(a)为 AD736,图1(b)表示 AD737。



(a) AD736



(b) AD737

图1 AD736/AD737

两者差别在于,AD736具有在片输出放大器;而AD737无在片输出放大器,输出电压为负,并具有功率降功能。若使 AD737的输出为正时,需外接一个放大器。

AD736、AD737的引脚2为高阻输入端。引脚1为低阻输入端,输入阻抗为 $8k\Omega$ 。AD736、AD737的变换误差很小,在一般应用中可不需加调整电路而直接使用。

当  $V_{IN}=0$  时,电源电流仅 $200\mu A$ 。因此,在使用中可用干电池做电源,特别是 AD737,因无在片放大器而又具有功率降功能,电源电流减小到 $40\mu A$ 。但随着输入信号的增大,电源电流亦增大。

AD736、AD737主要参数如表1所示

表1 AD736/AD737参数

| 型号     | 转换精度<br>( $mV \pm \%RD$ ) |             | 满量程电压<br>$V_{rms}$ | 频率特性<br>$f_{-3dB}$<br>MHz | 电源电压/<br>电流(V/mA)    | 备注  |
|--------|---------------------------|-------------|--------------------|---------------------------|----------------------|-----|
|        | 内部调整                      | 外部调整        |                    |                           |                      |     |
| AD736J | $0.5 \pm 0.50$            | $1 \pm 0.5$ | 0.2                | 0.46                      | $+2.8 \sim \pm 16.5$ | 有输出 |
| AD736K | $0.3 \pm 0.30$            | $1 \pm 0.3$ | 0.2                | 0.46                      | $-3.2 \sim \pm 16.5$ | 缓冲器 |
|        |                           |             |                    |                           | /0.16                |     |
| AD737J | $0.4 \pm 0.50$            | $1 \pm 0.2$ | 0.2                | 0.46                      | $+2.8 \sim \pm 16.5$ | 无输出 |
| AD737K | $0.2 \pm 0.30$            | $1 \pm 0.2$ | 0.2                | 0.46                      | $-3.2 \sim \pm 16.5$ | 缓冲器 |
|        |                           |             |                    |                           | /0.12                |     |

### 二 基本用法及实测特性

AD736的基本电路如图2所示。图2(a)为由引脚2输入,图2(b)为由引脚1输入时的基本接法。图中  $C_{AV}$  为平均电容。

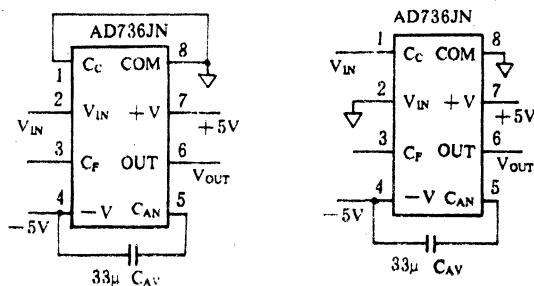


图2 AD736的基本电路

由引脚2输入且小信号输入时的实测频率特性如图3所示。从图中看到,当  $C_{AV} = 33\mu F$ 、满量程( $200mV_{rms}$ )的1%误差频率为 $30kHz$ 。而大信号输入时( $V_{IN} = 0.5V_{rms}, 1V_{rms}$ )的频率特性如图4所示。从图中看到,当  $V_{IN} = 0.5V_{rms}$  时,输出误差为 $-0.6\%$ ,当  $V_{IN} = 1V_{rms}$  时,输出误差为 $-1.3\%$ 。

引脚1输入且小信号输入时的实测频率特性如图5所示。图5曲线的测量条件是  $C_{AV} = 33\mu F$ ,  $V_{IN}$  分别取  $2mV_{rms}$ 、 $20mV_{rms}$  和  $200mV_{rms}$ 。满量程时  $f_{1\%}$  为 $90kHz$ ,带宽约为引脚②输入时带宽的3倍。

图6表示 $200mV_{rms}$ 输入满量程情况时,AD736的线性特性实测曲线。在 $2mV_{rms} \sim 200mV_{rms}$ 范围内,读值相



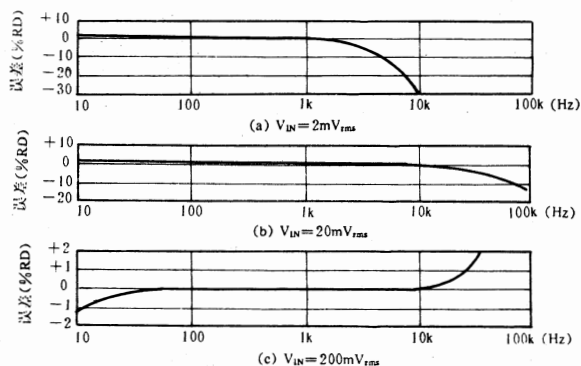


图3 AD736小信号输入频率特性

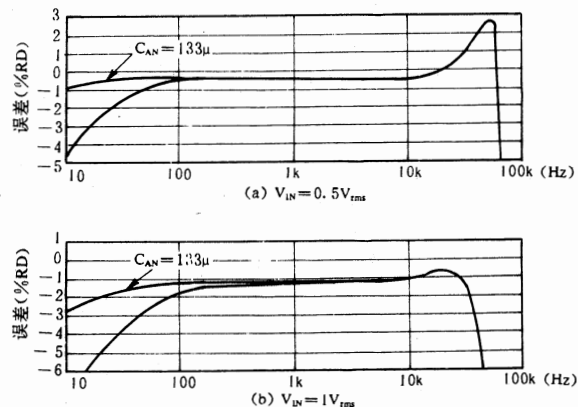


图4 AD736大信号输入频率特性

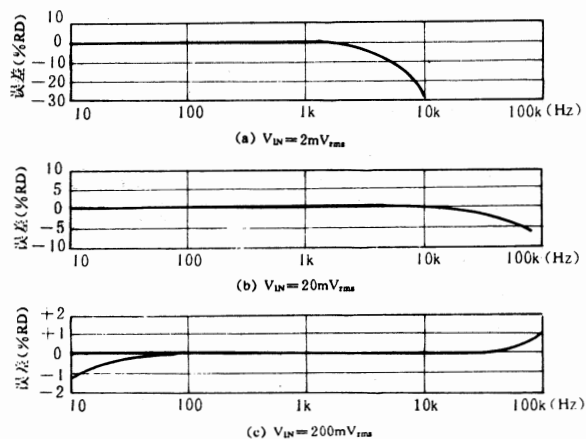


图5 AD736引脚①小信号输入频率特性

对误差在0.1%以内。

从上述实测特性来看,AD736是一种廉价且变换特性良好的集成电路。但当输入信号大于200mV<sub>rms</sub>时,误差将显著增大。

### 三 C<sub>AV</sub>的选择

平均电容 C<sub>AV</sub>的值直接影响 AD736的低频特性。C<sub>AV</sub>应选择较大的值,以使 AD736用于较低的频率。例如,选择 C<sub>AV</sub>=33μF, f=10Hz 时,直流误差为-1%。

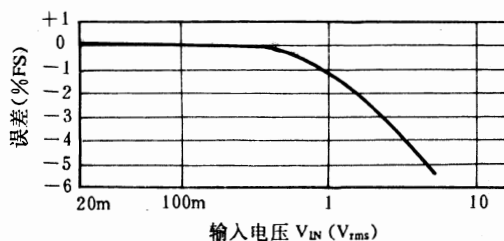


图6 AD736线性特性

与其他 RMS-DC 变换 IC 不同,输入电压 V<sub>IN</sub> 的大小不同,其低频特性亦不同。这是因为 C<sub>AV</sub>实际上与 AD736内部的二极管的两端相连,当输入信号大小不同时,二极管的内部电阻不同。当输入信号 V<sub>IN</sub>变小时,二极管内部电阻增大,从而使由 C<sub>AV</sub>决定的时间常数增大。

因此,当 V<sub>IN</sub>大于200mV<sub>rms</sub>, C<sub>AV</sub>=33μF 时,低频特性恶化。若改善低频特性,需选大的 C<sub>AV</sub>值。

C<sub>AV</sub>和平均误差的关系如图7所示。图7的测试条件是: V<sub>IN</sub>=200mV<sub>rms</sub>, C<sub>c</sub>=47μF, C<sub>F</sub>=47μF, V<sub>s</sub>=±5V。

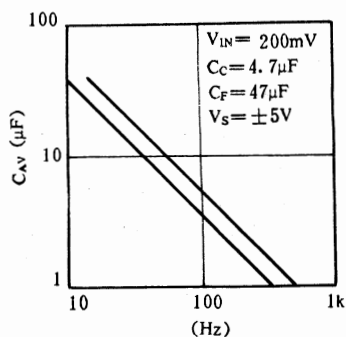


图7 C<sub>AV</sub>和平均误差关系

输入电平和 DC 误差的关系如图8。图8的测试条件为: V<sub>IN</sub>为正弦波输入,交流耦合, C<sub>AV</sub>=10μF, C<sub>c</sub>=47μF, C<sub>F</sub>=47μF, V<sub>s</sub>=±5V

减小交流误差即除去纹波电压的电路如图9所示。在引脚③(C<sub>F</sub>)和引脚⑥(V<sub>OUT</sub>)间跨接一个电容(C<sub>F</sub>=10μF),这时,高频端截止频率 f<sub>CH</sub>为:

$$f_{CH} = 1 / (2\pi \cdot 8k\Omega \cdot C_F) \text{ (未完,待续)}$$

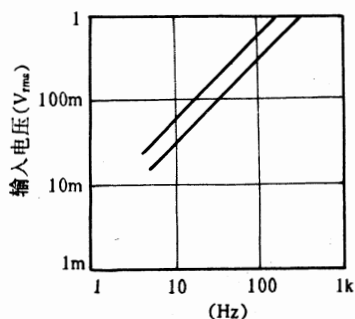


图8 输入电平和 DC 误差

# Intel 486 系列 CPU 综述

吴 胜 北京261信箱监测中心(102101)

由于计算机技术的飞速发展,使386、486档次的微机已逐步走向家庭,现今市场上 IBM 386 VGA 彩显+硬盘+双高密软驱原装微机售价也只八九千元,而配置廉价的386、486家用电脑售价也不过五六千,有关 CPU 性能一般用户尚不甚了解,本文引出了 iCOMP 的概念,从 Intel 公司芯片的自我评价角度,对486各型 CPU 做一简单分析,供广大 PC 机用户参考。

另外提醒一下,现在购机,至少应购买286机,其售价并不比8088/8086档次高,而性能好得多,至于中产阶层应考虑386/33以上档次微机,以考虑系统升级及为日后的软件应用留下余地。

## 一 Intel 486、386系列 CPU 比较

1993年四季度,Intel 公司在市场上大力推广新一代芯片 Intel 486 DX3。

至此,Intel 486家族已有四个成员和 Intel Over-Drive 升级芯片。为何在 Intel 推出第五代 CPU 奔腾(Pentium)后,仍不遗余力地发展486系列,其中自有深意。

一般认为,486是一个386加上一个浮点运算器387,再加上片内高速缓存 Cache。实际情况并不这么简单。

1. 486在 Intel CPU 历史上首次采用了 RISC 技术,即精简指令系统。这并不意味着486与386不兼容,因为486的指令并未精简,强调的只是 RISC 技术。采用 RISC 技术的目的,是要使486达到一个时钟周期执行一条指令。486 CPU 一般均达到了此设计指标,平均每一个机器周期执行1.2条指令。

2. 486 CPU 将数学协处理器和 Cache 及 Cache 控制器一块集成到片内,极大地提高了 CPU 的处理速度。如图1所示,

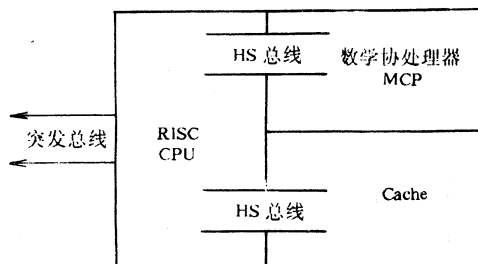


图1

数学协处理器与 RISC CPU 之间以及 Cache 与 RISC CPU 之间均采用高速片内数据总线 HS-BUS 进行数据传送。其中数学协处理器是387DX,比通常的

387协处理器增加了正弦(Sin)和余弦(Cos)两条函数指令。387DX 与 RISC CPU 之间的高速数据总线宽度为64字节,而 Cache 同 RISC CPU 之间则为128字节。如此带宽的数据交换通道是板上协处理器和 Cache 所无法企及的。目前386主板上的 Cache 与 CPU 的数据交换通道一般是32位,最大可以到64位。

3. 486采用了突发总线 Burst Bus 同 RAM 进行高速数据交换。如图1所示。通常的传统交换方式是取得一个地址,交换一个数据,再取指,再交换一个数据。而采用突发总线后,每取一次地址后将交换此地址及以后地址空间的一组连续数据。这种技术很适合网络应用,因为网络上经常要进行一连串地址空间的数据传送。从表1可看到,很明显,突发总线的传送效率极大地提高了486 CPU 的性能。

表一

| CPU          | 386SX-20 | 386DX-33 | 486SX-25 | 486DX-50 |
|--------------|----------|----------|----------|----------|
| 与 RAM 间数据传输率 | 20MB/S   | 66MB/S   | 80MB/S   | 160MB/S  |

基于以上三点优势,在相同时钟频率下,486 CPU 的处理速度一般比386快2-3倍。主频为25MHz 的486SX-25,在运行 Microsoft C 6.0 和 AutoCAD 11.0等8种典型应用程序时,其运行速度平均也是时钟频率为40MHz 的386DX-40的1.22倍。

## 二 486芯片性能介绍

目前的 Intel 486 系列共有 486 SX, 486DX, 486DX2, 486DX3四大类及一种 Over Drive 升级芯片。

### 1. 486SX CPU

Intel 486SX CPU 是486 系列的初级产品,不具备数学协处理器,其结构简图如图2所示。486SX 兼有486的性能和386的价格两大优势。目前486SX 有16MHz, 20MHz, 25MHz 和33MHz 等几种产品。由于486SX 在生产过程中不用测试数学协处理器,成品率高,采用塑料包装,不一定就比386DX 贵。

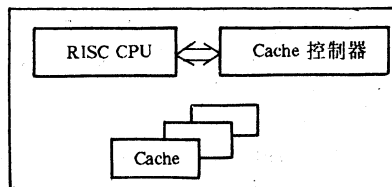


图2

## 2. 486DX CPU

486 DX 系列的结构简图如下

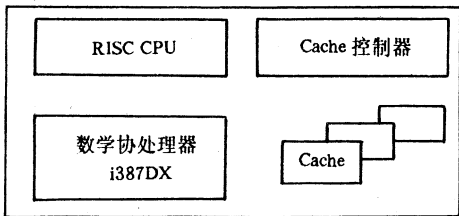


图3

其特点是具备了486系列的基本功能,是486的典型芯片。它比486SX增加了片上数学协处理器的浮点运算功能,但价格较贵。486DX的时钟主频一般较高,有25MHz,33MHz及50MHz等几种。

## 3. 486DX2 CPU

486DX2在486DX基础上增加了倍速总线接口(Clock Doubling Bus Interface),其结构简图如下

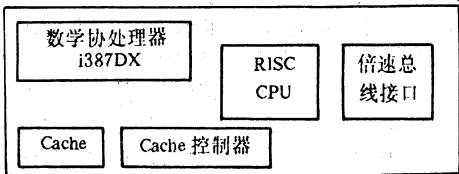


图4

它采用了倍速技术(Speed Doubling),使CPU的执行速度两倍于系统总线的速度。目前486DX2的时钟主频主要有50MHz和66MHz两种。也就是说50MHz和66MHz的两种486DX 2,其系统总线分别应是25MHz和33MHz。

## 4. 486DX3 CPU

486DX3是486家族中的最新成员。与486DX2相似,其内部CPU的工作频率是外部的三倍,其结构简图如下

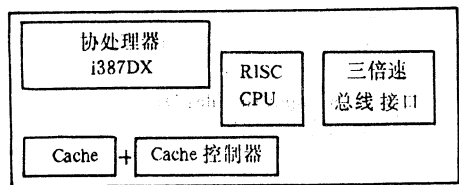


图5

486DX3在性能上几乎与Pentium(奔腾)CPU相当,而价格却低得多。

据称,DX3有两种,一种是主要用于便携机市场的486DX3-75,另一种是高性能的486DX3-100。

## 5. Intel Over Driver

Intel Over Driver是一般用户不大熟悉的升级芯片。这也是486系列与386系列CPU的重要差别之一。

在486产品中,Intel提供了一种升级办法,即采用Over Driver升级芯片,用比较少的花销得到更高的系

统性能。

在系统升级的处理方法上,一般厂家是用加插板子的办法,而Intel则在486系列之后采用了廉价的单片升级办法,即在原486CPU的旁边加插一块Over Driver芯片。Over Driver从原理上说也是一个CPU,它采用Intel的倍速技术,能将原CPU处理速度提高近一倍。表2列出了几种CPU在加插Over Driver的前后性能比较,其测试基准为SPEC int89。

表2 CPU加插Over Driver前后性能比较

| 型号            | 386<br>SX-25 | 386<br>DX-33 | 486<br>SX-20 | 486<br>SX-25 |
|---------------|--------------|--------------|--------------|--------------|
| 未插Over Driver | 4.2          | 8.0          | 10.7         | 12.9         |
| 插Over Driver后 | — *          | —            | 18.1         | 20.3         |

| 型号            | 486DX-33 | 486DX-50 |
|---------------|----------|----------|
| 未插Over Driver | 17.4     | 28.5     |
| 插Over Driver后 | 27.0     | —        |

\* 注意:表中横杠表示不能用Over Driver升级

目前,Over Driver可以应用于16MHz、20MHz和25MHz的486SX系统以及25MHz、33MHz的486DX系统,对于50MHz的486DX和486DX2,Intel不适用。

另外,还有一种功能更强的Over Driver处理器芯片,不象前一种是插在CPU旁边,而是直接替换原有486CPU,得到更强的处理功能,但由于这种方式拔出和插入芯片均需45kg以上的力量,一般用户难以企及,应用尚不广泛。

## 三 CPU性能比较指标:iCOMP

在市场上,各厂家电脑产品纷繁复杂,其采用的CPU,有486、386之分,有SX、DX、DX2、DX3各档,有Over Drive和倍速技术及浮点运算功能,有不同的时钟频率,即使是从事计算机专业的人员也很难一下弄明白究竟哪一种CPU更适合自己的性能要求,何况大量不懂得技术细节的广大PC机购买者。

例如,主频为50MHz的486DX和486DX2,哪一个处理速度更快一些?一般用户可能会以为是DX2比DX好,实则不然,50MHz的486DX2由于采用倍速技术,使其CPU达到了50MHz,而CPU 486DX-50的内外部均为50MHz,因此其处理速度更快。

为此,计算机界目前提出一种简单、单一的指标来描述CPU的性能,这就是iCOMP,它是英文Intel COMparative Microprocessor Performance的缩写。iCOMP指标用来反映一种Intel CPU相对于另一种Intel CPU的性能差异,其指标设定不光考虑目前的桌面计算机系统应用,而且考虑到3~5年软件技术的提高带来的应用方面的变化。

表3 iCOMP使用测试基准和权重

| 应用类型    | 测试基准            | 权重  |
|---------|-----------------|-----|
| 16位整数运算 | ZD labs CPU mix | 67% |
| 16位浮点运算 | Whetstone       | 2%  |

|         |                    |     |                        |     |
|---------|--------------------|-----|------------------------|-----|
| 16位浮点运算 | ZD lab CPU mix     | 1%  | i386 SX-25             | 39  |
| 32位整数运算 | SPEC int92         | 25% | i386 SL-25             | 41  |
| 32位浮点运算 | SPEC fp92          | 5%  | i386 DX-25             | 49  |
| 16位视频运算 | ZD labs/SPEC int92 | 0%  | i486 SX-16             | 63  |
| 32位视频运算 | ZD labs/SPEC int92 | 0%  | i486 SX-16+Over Driver | 132 |
| 16位图形运算 | ZD labs/SPEC int92 | 0%  | i386 DX-33             | 68  |
| 32位图形运算 | ZD labs/SPEC int92 | 0%  | i486 SX-20             | 78  |
|         |                    |     | i486 SX-20+Over Driver | 166 |
|         |                    |     | i486 SX-25             | 100 |
|         |                    |     | i486 SX-25+Over Driver | 231 |
|         |                    |     | i486 DX-25             | 122 |
|         |                    |     | i486 SX-33             | 136 |
|         |                    |     | i486 DX-33             | 166 |
|         |                    |     | i486 DX2-50            | 231 |
|         |                    |     | i486 DX-50             | 249 |
|         |                    |     | i486 DX2-66            | 297 |
|         |                    |     | Pentium-60             | 510 |

为此,选择了适用于 DOS 应用的16位整数运算,适合 CAD/CAM 应用的16位浮点运算,适用于 UNIX 应用的32位整数和浮点运算,适用多媒体应用的16位、32位视频运算及16位、32位图形运算等8种应用类型,分别赋以不同的权值,用不同的测试基准对不同的 CPU 进行测试。

然后以486SX-25作为一个基准,用下面的公式计算 CPU 的 iCOMP 值:

$$iCOMP = 100 \cdot \left( \frac{BM_1}{Base - BM_1} \right)^{P_1} \cdot \left( \frac{BM_2}{Base - BM_2} \right)^{P_2}$$

...

其中 Base-BM<sub>j</sub> 代表486SX-25在第j个指标下的基准测试值,BM<sub>j</sub> 代表被测 CPU 在第j个指标下的基准测试值,P<sub>j</sub> 为第j个指标的权值,计算结果即为被测 CPU 的 iCOMP 值。

由此公式,可以得到 Intel CPU 家族中部分 CPU 的 iCOMP 值,如表4所示。

表4 部分 CPU 的 iCOMP 值

|            |    |
|------------|----|
| i386 SX-16 | 22 |
| i386 SX-20 | 32 |

## ·回音壁·

# 也谈 WPS 文件阅读器

张国栋 中国科技大学无线电系(230027)

“电子与电脑”93.10刊登了一篇名为“WPS 文件阅读器”的文章,作者提出了一种在 DOS 状态下查看 WPS 文件内容的方法,不失为一种较方便的方法。

但当 WPS 文件设置密码后,此文章所提出的方法便不能正常显示文件内容。

本人对 WPS 文件进行分析后,得出了文件密码和文件内容的关系,文件的密码存放在偏移量为 02DDH 到 02E4H 的8个字节中,若无密码则 02DDH 内容为 00H。文件内容存放在偏移量为 0400H 之后的区域中,当无密码时只要将文件内容依次读出即可,原文即是如此方法,而有密码时,必须将文件内容与密码区各字节进行循环“异或”操作,才能得到正常的内容代码。

根据以上原理,本人编写了对有、无密码的 WPS 文件均适用的显示程序。此程序在386微机上调试成

有了以上的 iCOMP 指标,一般用户就可对 Intel 各种 CPU 的性能有一个大概的了解。

随着计算机应用的深入,根据不同应用类型的比重发生变化,比如将来由于多媒体应用而使32位视频运算所占权值不再是0%,iCOMP 也会随之发生变化,但这种变化是渐进的。Intel 承诺向所有的用户公开 CPU 的 iCOMP 值。但用户不可以仅仅根据 iCOMP 值来选购 PC 系统。因为 iCOMP 值只代表了 CPU 的性能,而不同厂家的 PC 系统,由于结构设计不尽相同,其系统的性能价格比才是最重要的。

功。

```
#include<stdio.h>
#include<string.h>
#include<stdlib.h>
main(int argc,char * argv[])
{
    FILE * fp;
    int i,num;
    char password[8],file[8],fname[20];
    if(argc<2){
        printf("Please input the file,");
        gets(fname);
    }
    else
        strcpy(fname,argv[1]);
    if((fp=fopen(fname,"rb"))==NULL)
    {
        puts("Cannot open the file!");
        exit(1);
    }
    fseek(fp,0x2dd,0);
    fread(password,1,8,fp);
    fseek(fp,0x400,0);
    for(num=fread(file,1,8,fp);num=fread(file,1,8,fp);num++)
        (下转47页)
```