

# VAX/VMS

## 系统的使用 及开发

匡松 编著

电子科技大学出版社

TP316  
7148

150040

# VAX/VMS系统的使用及开发

匡 松 编著

电子科技大学出版社

1992

## 内 容 提 要

本书介绍了VAX/VMS操作系统概念、内容及特点。系统介绍了VAX/VMS的DCL命令语言、终端操作、用户工作环境、文件与目录的管理、文件保护机制、EDT文本编辑模式及编辑命令、程序开发技术、库的管理与使用、DEBUGGER符号调试程序、改进用户界面以及命令过程等内容。

本书内容全面而丰富,列举了大量实例,具有很强的实用价值。本书可作为从事VAX系列计算机的系统管理、软件开发的工程技术人员的培训教材或工作参考书,也可作为大专院校使用VAX机的学生的教材。

## VAX/VMS系统的使用及开发

匡 松 编 著

\*

电子科技大学出版社出版发行

(中国成都建设北路二段四号)

南充县印刷厂激光照排胶印

\*

开本 787×1092 1/16 印张 17.625 版面字数 451 千字

版次 1993年2月第一版 印次 1993年2月第一次印刷

印数 1~3000册

中国标准书号 ISBN 7-81016-483-X/TP·39

[川]016(15452.201) 定价:8.75元

## 序

当前,随着DEC 公司各种档次的系列机的推出,开放系统战略性地“破门而出”了。但这似乎并不意味VAX 机专用操作系统VAX/VMS 历史使命的完结,在拥有数千用户的中国尤其如此。

自八十年代起,我国开始选择、引进VAX 机,十年来应用、开发VAX/VMS 的实践,使我们造就了一大批驾驭该系统的专业人才。作者当然应属于这个范围之内。毋庸置疑,也还有不少管理使用VAX 机的技术人员,特别是“后来人”和“半路出家”的系统管理人员以及程序开发人员需要较为系统地学习和掌握VAX/VMS 的应用开发思想和手段,以便更好地发挥系统的潜能。

由此而论,作者用心血著成的《 VAX/VMS 系统的使用及开发》一书当具有实实在在的使用价值。

本书介绍了VAX/VMS 操作系统概念、内容及特点。系统介绍了VAX/VMS 的DCL 命令语言、用户工作环境、文件与目录的管理、文件保护机制、EDT 文本编辑模式及命令、程序开发技术、库的管理与使用、DEBUGGER 符号化调试程序、改进用户界面以及命令过程等内容。当是从事系统管理、程序开发工作的人员以及大专院校学生和各层次VAX 机用户的参考书。

当然,近年来,一些用户单位也编著了一批类似的工具书并在各种范围内发挥着作用,但是公开出版这种内容的专著尚属少见。所以本书的出版是应该祝贺的。

作者千里寄函撰写序言,自愧不能担此重任,然执意推辞又恐冷了青年朋友的心,便匆匆奉此,权作序言吧。

张 明

一九九二年十一月

CS559/1206

## 前 言

VAX系列计算机是世界上著名的小型机系列,VAX/VMS亦是最优秀的操作系统之一。在我国拥有众多的VAX机用户。

为了帮助用户系统地学习、理解和使用VAX/VMS,作者参考了众多的资料并根据自己近几年来从事VAX机系统管理、开发与应用的心得、体会及经验,认真编著了《VAX/VMS系统的使用及开发》一书。

本书共分十三章。第一章介绍了VAX/VMS操作系统的内容、层次结构、功能及特点。第二章至第五章介绍了系统的进入与退出、终端操作、DCL命令语言和用户工作环境。第六章至第八章介绍了文件管理、文件保护和EDT编辑程序。第九章至第十一章介绍了程序开发技术以及系统软件资源的综合应用、库文件的管理与使用、DEBUGGER符号调试程序。第十二章至第十三章介绍了改进用户界面、命令过程的设计及应用。书末还给出了四个附录,列出了DCL命令一览表、词法函数的格式及功能提要等内容,供读者查阅。

本书内容深入浅出,尤其注重实用性。为了便于理解和使用,书中列举了大量的应用实例。

本书的出版得到了西南石油学院科研处的大力支持,承蒙电子科技大学出版社社长杨旭明教授全面审阅并提出了宝贵的修改意见,中国计算机用户协会DEC系列机分会秘书长、《VAX通讯》主编张明先生在百忙中热情为本书作序,在编著此书的过程中,多次得到了西南石油学院高卫东副教授、易宗富副教授真诚而热情的鼓励,在此,作者谨向他们深表谢意。

由于作者水平有限,对VAX/VMS理解和消化不够,书中难免有错误和不足之处,敬请读者不吝指正。

作 者

1992年11月

# 目 录

|     |                       |         |
|-----|-----------------------|---------|
| 第一章 | VAX/VMS操作系统概述 .....   | ( 1 )   |
| 1.1 | VAX/VMS的内容及层次结构 ..... | ( 1 )   |
| 1.2 | VAX/VMS的功能及特点 .....   | ( 2 )   |
| 第二章 | 进入系统和退出系统 .....       | ( 6 )   |
| 2.1 | 进入系统 .....            | ( 6 )   |
| 2.2 | 退出系统 .....            | ( 8 )   |
| 第三章 | 终端操作 .....            | ( 9 )   |
| 3.1 | 屏幕与键盘 .....           | ( 9 )   |
| 3.2 | 终端操作特性的设置及改变 .....    | ( 12 )  |
| 3.3 | 特殊功能键与ESC控制序列 .....   | ( 18 )  |
| 第四章 | DCL命令语言 .....         | ( 23 )  |
| 4.1 | DCL命令的功能及种类 .....     | ( 23 )  |
| 4.2 | DCL命令的格式 .....        | ( 24 )  |
| 4.3 | DCL命令的特点 .....        | ( 26 )  |
| 4.4 | 系统对DCL命令的响应 .....     | ( 28 )  |
| 4.5 | 部分DCL命令的使用 .....      | ( 29 )  |
| 4.6 | DCL命令的扩充 .....        | ( 32 )  |
| 第五章 | 用户工作环境 .....          | ( 41 )  |
| 5.1 | 系统对用户工作环境的定义 .....    | ( 41 )  |
| 5.2 | 显示工作环境信息 .....        | ( 45 )  |
| 第六章 | 文件管理 .....            | ( 54 )  |
| 6.1 | 文件标识 .....            | ( 54 )  |
| 6.2 | 文件的目录结构 .....         | ( 60 )  |
| 6.3 | 目录与文件的管理 .....        | ( 61 )  |
| 第七章 | 文件保护 .....            | ( 74 )  |
| 7.1 | 基于UIC的保护 .....        | ( 74 )  |
| 7.2 | 基于ACL的保护 .....        | ( 78 )  |
| 第八章 | EDT编辑程序 .....         | ( 87 )  |
| 8.1 | EDT的调用与退出 .....       | ( 87 )  |
| 8.2 | EDT工作缓冲区 .....        | ( 90 )  |
| 8.3 | EDT行编命令 .....         | ( 92 )  |
| 8.4 | EDT屏编功能键 .....        | ( 107 ) |
| 8.5 | EDT键定义 .....          | ( 116 ) |
| 8.6 | EDT的文件恢复 .....        | ( 121 ) |
| 第九章 | 程序开发 .....            | ( 122 ) |
| 9.1 | 程序开发步骤及方法 .....       | ( 122 ) |

|        |                               |       |
|--------|-------------------------------|-------|
| 9.2    | BASIC、FORTRAN及MACRO程序开发 ..... | (125) |
| 9.3    | 多种语言程序之间的相互调用 .....           | (138) |
| 9.4    | 系统服务、运行时间库以及实用程序的应用 .....     | (149) |
| 第十章    | 库文件的管理与使用 .....               | (172) |
| 10.1   | 库的概念、类型及结构 .....              | (172) |
| 10.2   | 库的管理与使用 .....                 | (173) |
| 10.3   | 帮助库的建立与使用 .....               | (176) |
| 第十一章   | 符号调试程序 .....                  | (180) |
| 11.1   | 符号调试程序的调用 .....               | (180) |
| 11.2   | 调试命令概述 .....                  | (181) |
| 11.3   | 调试程序的主要命令介绍 .....             | (184) |
| 11.4   | 程序调试举例 .....                  | (191) |
| 第十二章   | 改进用户界面 .....                  | (198) |
| 12.1   | 逻辑名 .....                     | (198) |
| 12.2   | 符号命令(命令同义词) .....             | (203) |
| 12.3   | 键定义 .....                     | (205) |
| 第十三章   | 命令过程 .....                    | (209) |
| 13.1   | 命令过程的建立与编排 .....              | (209) |
| 13.2   | 命令过程中的运算操作与符号替换 .....         | (212) |
| 13.3   | 命令过程的输入输出 .....               | (217) |
| 13.4   | 词法函数的使用 .....                 | (224) |
| 13.5   | 命令过程中的文件I/O操作 .....           | (231) |
| 13.6   | 命令过程的执行 .....                 | (237) |
| 13.7   | 命令过程的应用 .....                 | (240) |
| 附录A    | DCL命令一览表 .....                | (261) |
| 附录B    | 词法函数的格式及功能提要 .....            | (268) |
| 附录C    | ASCII代码表 .....                | (273) |
| 附录D    | ASCII符定义 .....                | (274) |
| 主要参考文献 | .....                         | (275) |

# 第一章 VAX/VMS操作系统概述

VAX/VMS (Virtual Address eXtension/Virtual Memory System) 是主要配置于32位超级小型机VAX-11系列机上的操作系统,由美国DEC (Digital Equipment Corporation) 公司研制,产生于70年代中期。VAX/VMS 是一种性能优越的虚拟存贮操作系统。

下面简要介绍VAX/VMS操作系统的内容、层次结构、主要功能及其特点。

## 1.1 VAX/VMS 的内容及层次结构

VMS是一种功能强、效率高、软件内容丰富的操作系统。经安装完成之后的实际操作系统可分成系统文件和库文件这两大类。其中所包含的内容主要有:系统引导程序、设备驱动程序、虚存管理及调度程序、网络软件、程序开发工具、分析工具、文本编辑工具、各种实用程序、用户环境测试包以及完成基本文件管理所需的文件、系统管理文件、队列管理文件、帮助文件、系统参数文件、转贮文件、描述系统信息的文件等。在VAX/VMS中还有很多支持系统工作的必需的软件,如条件处理程序、系统服务程序、软件中断、硬件中断、出错处理、计时器支持服务程序、进程创建与删除、I/O系统服务等。这些内容以文件的形式分布在磁盘上操作系统目录结构中的多个目录下。

从用户角度往系统内看,VAX/VMS操作系统具有以下层次结构:

### 1. 用户层

这一层主要有各种语言的编译系统、EDT编辑程序、连接程序、库以及操作系统提供的大量实用程序——即面向用户设计的公用程序,如复制、记错、安装、帮助等。操作系统在这一层上运行时称之为“用户方式”。

### 2. 管理层

这一层主要是DCL(Digital Command Language)命令解释程序。这是VAX/VMS操作系统与用户之间的软接口——即操作系统和用户对话的主要工具。操作系统在这一层上运行时称之为“管理方式”。

### 3. 执行层

这一层主要是RMS记录管理服务,它是文件管理的核心部分,其功能是既对用户软件进行文件式管理,又对操作系统本身的资源进行文件化管理。操作系统在这一层上运行时称之为“执行方式”。

### 4. 核心层

这一层主要包括控制进程及系统软件硬件资源的系统服务程序模块。它们完成基本的操作系统功能,提供进程间的通讯、协调I/O等操作。操作系统在这一层上运行时称之为“核心方式”。

从第四层再往里就是操作系统围绕机器硬核的软核部分。其内容有管理进程的调度模块、管理存储器的交换模块和调页模块、设备驱动程序、基本数据基、I/O子系统以及系统响应软件硬件中断请求并完成实时处理请求的中断子系统。



上述四种方式分别决定了在系统中的活动能力。这样在特权较低的访问方式下运行的程序不会侵犯和影响比它特权较高的数据代码。这体现了VMS操作系统具有很好的保护机能。

## 1.2 VAX/VMS的功能及特点

VMS操作系统调度在系统中运行的程序、监控和协调整个系统的功能,管理物理存储器的有效使用。它提供了十分大的虚拟地址空间;有很好的适应性、兼容性和安全性。下面简要介绍VMS在存储管理、进程调度、RMS记录管理服务等几个方面的实现及特点。

### 1. 存储管理

VAX-11的存储体系是由超高速缓存、主存、磁盘等硬设备和存储管理机构所组成。VAX/VMS采用的是虚存管理技术。所谓虚存管理技术是在较小的物理内存空间中容纳多个用户,且每个用户的编程空间不受实际分配的物理空间的限制。它的实质就是将辅存(又叫外存。如磁盘等)当成虚拟的内存来统一调度和管理,从而使得从每个用户的角度来看,似乎都有一个自己独占的只受地址位数限制的极大的寻址空间。VAX机字长32位,可以提供的虚拟空间为 $2^{32}$ 字节,即4.3GM。在VMS中,用户可以完全不必为自己的程序大小担忧。

在VMS中,虚拟内存(辅存)上的调度、管理是由页面管理(Pager)和交换管理(Swapper)协同完成的。Pager负责单个页面(每个页面为512个字节)在内、辅存之间的交换,而Swapper负责整个进程的工作集在内、辅存之间的交换。Pager是VMS操作系统中提供的专门用于处理页故障的例行服务程序,它由系统中所有的进程共享,且总是在发生页故障的进程关联下运行于核心方式。由于每个进程驻内存的页面数是有限制的(这些页面叫驻内集,也叫工作集),因而在进程的运行过程中,难免会出现欲访问的页不在工作集中的所谓页故障。当出现缺页故障时,Pager获得控制,由它来负责将所需要的页面调入内存以及在必要时将一个修改过的驻内存页面写回辅存,以便腾出空间装入新的页面。

当一进程的工作集空间已满而又有新的页面要进入时,必须先淘汰它自己的一个页面,即它的新的页面的进入必须替换其自己一个旧的页面,以确保工作集不会超出规定的限额。

一进程整个工作集的换入/换出叫“交换”,它由Swapper负责实施。Swapper是一个独立的进程,它有自己的进程关联且优先级总是保持在16级不变。导致一进程的工作集整个地被调出的原因,是它由于某种原因被阻塞而内存空间十分紧张。Swapper也随时将优先级最高的可执行进程调入内存并随时维护内存中有一定数量的自由空间待用。

正是由于有了Pager和Swapper,才能得外存变成内存的“延伸”部分(虚拟的内存)成为可能。这就是在采用了虚存管理技术的操作系统中把外存叫做辅存的原因。这时的外存已不是简单的外部存储器的概念,而是起到了补充、辅助满足用户对内存空间的需要作用。

### 2. 进程调度

所谓进程是指执行一个映象的总环境(软件环境和硬件环境),是一个能被调度执行的基本实体。在VMS系统中有多个进程共存。进程有三种基本状态:执行、可执行(即就绪)和等待,且可在这三种状态之间转换。处于可执行和等待状态的进程既可在内存也可在辅存,如图1.1所示。

这三种基本状态又有以下调度状态:

#### (1) 执行状态 —— CUR

表示该进程已占用处理机,正在CPU中运行。

### (2)驻内可执行状态 —— COM

这种状态是唯一能被调度到执行状态的状态。处于这种状态的进程已具备了执行所需的各种条件,只待获得CPU而投入执行。

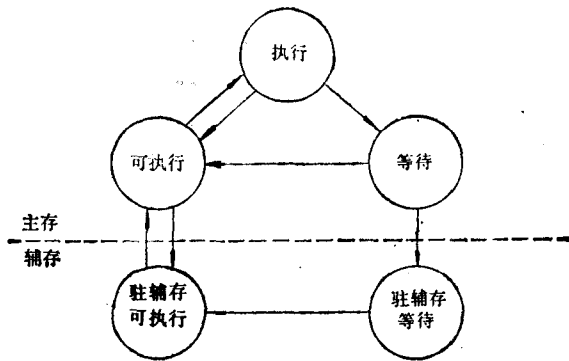


图 1.1

### (3)非驻内可执行状态 —— COMO

这一状态是驻在辅存上的可执行状态,它不能被调度执行,只有先转换成COM状态,然后才有可能获得竞争处理机的机会。

### (4)公共事件标志等待 —— CEF

这一状态涉及驻在内存中的多个队列。它是由于等待一个或多个公共事件标志(如进程间通讯等等)而进入等待队列的。不同的公共事件排在不同的等待队列中。这种队列一旦事件满足后又被换出内存而进入COMO 队列。

### (5)驻内局部事件等待状态 —— LEF

这种等待状态大多是由于调用系统服务一时得不到满足而从CUR 状态转换来的。

### (6)非驻内局部事件等待状态 —— LEFO

这种等待状态是由LEF 状态被换出内存而来的。

### (7)驻内睡眠状态 —— HIB

这种状态是从CUR 状态调用“睡眠”系统服务转换得来。它可以通过“唤醒”系统服务或删除进程等事件转换成COM 状态;还可以被换出内存进入HIBO 状态。

### (8)非驻内睡眠状态 —— HIBO

这种状态是由HIB 状态被换出内存得到。

### (9)驻内挂起状态 —— SUSP

“挂起”系统服务使CUR 状态转换成SUSP 状态。

### (10)非驻内挂起状态 —— SUSPO

由SUSP 换出变成这种状态。

### (11)页故障等待状态 —— PFW

由于进程所需的页面不在内存,需从虚空间(辅存)读入时,则该进程从CUR状态变为PFW 状态。页故障消除后可能换成COM 状态或COMO 状态。

### (12)自由页等待状态 —— FPG

这是一种资源等待。当进程需要增加若干页工作集,而内存可用页面(自由页)又不够分配时,则进程从CUR 状态转换成FPG 状态。自由页满足后转换成COM 或COMO 状态。

### (13)冲突页等待状态 —— COLPG

当多个进程同时出现“共享页面”的页故障时,第一个出现这种页故障的进程从CUR 状态转换成PFW 状态;而以后再出现这种页故障的进程,统统从CUR 状态转换成COLPG 状态。

### (14)杂类等待状态 —— MWAIT

杂类等待状态就是CUR 状态的进程由于某种资源一时得不到满足或未获得某一互斥信号量的使用权而转换成MWAIT状态。由于资源等待和互斥信号量等待种类繁多,所以统归结为杂类等待队列。当事件得以满足后转换成COM 或COMO 状态。

VAX/VMS 的进程调度使用的是可剥夺式优先权调度策略。系统内的进程被分成普通进程和实时进程这两大类。VMS 设有0~31级优先权。最高优先权为31级,最低优先权是0级。规定0~15级用于普通进程,16~31级则分配给实时进程,如图1.2所示。

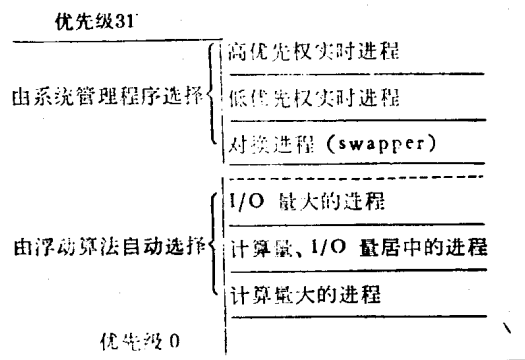


图 1.2

实时进程按照严格的剥夺式优先级算法进行调度。也就是说,一个优先权级更高的进程就绪,将剥夺一个正在执行的优先权级较低的进程的CPU。

若当前无实时进程时,调度程序便在普通进程之间分配处理机的服务时间,并且从高一级优先权的进程开始分配。对相同优先权的进程则首先执行排在队列前面的进程。这就形成了时间片轮转法,不过调度是按优先权进行的。

普通进程在执行过程中,其优先权根据系统事件和调度次数而动态变化,如访问磁盘、输入输出、等待消息等都称为系统事件。每一种系统事件对应一个优先数增量。当由于该事件的出现导致一个进程就绪时,系统将此新就绪进程的优先数置为它自己的基优先数(在VMS中,一进程的基优先数是在进程创建时确定的)加上与该事件对应的增量。普通进程每被调度执行一次时,它的优先数便下降一个数量,直至达到基优先数,但决不会低于它自己的基优先数。于是,如此便获得均衡使用处理机的效果。

### 3. RMS 记录管理服务

VMS 中的记录管理服务子系统RMS 负责文件的组织和存取。在VMS 中,文件有以下三种组织形式:

#### (1)顺序文件组织

这是最简单的文件组织形式,各个记录顺序写入并决定各记录在文件中的顺序。在顺序文件中,每个记录相对于其它记录来说其位置是固定不变的。因此,记录之间不允许插入新记录,要加入新记录时只能写在文件末尾。当要修改某个记录时,修改后的记录 and 原记录必须在格式和长度上完全相同。

#### (2)相对文件组织

相对文件是由一系列有固定长度的单元组成。用户可以说明单元的长度。这些单元从1到n连续编号,即单元号。每个单元可以存放一个记录,也可以是空的(如删除记录后为空或初始写文件时预留的空单元等)。空单元可以出现在文件中的任何地方,它们不必是连续的。一个单元的编号代表了该单元与文件起始点的相对位置。单元编号又称相对记录号,常常就简称为记录号。用户可用单元号来识别和访问记录。修改文件记录时,只要长度不超过单元大小就可任意改变记录长度。如果用户要求按记录号随机访问记录和删改记录,则可采用相对文件组织形式。

### (3)索引文件组织

索引文件组织是一种能按记录中的数据寻找记录的组织形式,即按记录中关键字的内容查找记录。

RMS 完全控制了索引文件记录位置的安排。文件记录的关键字的存在支配了这种安排。

关键字是在每个索引文件记录中出现的一个字节串。它可以使用六种不同的关键字类型:字符串、有符号的15位整数、无符号的16位二进制数、有符号的31位整数、无符号的32位二进制数以及压缩十进制数。索引文件是文件组织形式中唯一的一种以数据而不是以地址进行存取的文件形式。关键字的位置和长度在所有记录中是相同的。

对于上述每一种文件组织形式, RMS 都支持定长和变长两种记录格式。其中对顺序文件和相对文件, RMS 还支持一些特殊的记录格式,如带有固定部分的变长记录等格式。RMS 提供了三种存取方式:顺序存取、用关键字或相对记录号随机存取和按记录的文件地址存取。另外, RMS 还提供了一种实现块I/O操作的方法以满足用户的特殊需要。

上面介绍了有关VMS 操作系统的几个重要功能。在进程通讯及I/O方面, VAX/VMS 也有其特点。VMS I/O 系统提供了I/O 队列管理模块、设备驱动程序、I/O 结果传送子程序来控制 and 协调输入输出;系统中多个相关的进程往往需要同步执行、互相发送消息或共享公共数据等。VMS 为此提供了公共事件标志、信箱、网络、共享存贮区和共享文件等多种通讯工具及技术来有效地实现进程之间的通讯与同步。

## 第二章 进入系统和退出系统

用户在使用VAX/VMS 系统之前,必须取得系统的认可。系统管理员将在系统中为用户建立帐户,设置必要的工作环境,并提供注册所用的用户名和口令。为了使用系统资源进行应用开发工作,用户首先必须完成注册过程,即通过人机对话输入用户名和口令进入系统。

### 2.1 进入系统

VAX/VMS 是一种交互式操作系统。为了进入系统。用户按下列步骤操作:

1. 打开终端电源。
2. 稍等片刻,在键盘上按RETURN (回车)键。
3. 当屏幕上出现“Username:”提示时,输入用户名并按回车键。
4. 当屏幕上出现“Password:”提示时,输入口令并按回车键。

例如,若一用户注册所用用户名是USERGAS,口令为MAKEWORK,其人机对话操作如图2.1所示:

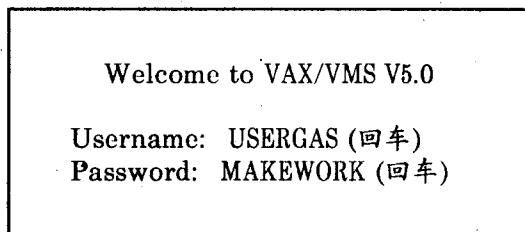


图 2.1

输入口令时,屏幕上不显示出所输入的口令串。如果输入的口令正确,用户将成功地进入系统。如果输入的口令(或用户名)有误,屏幕上立即出现以下信息:

User authorization failure (用户授权失效)

此行信息通知你进入系统失败。在此情况下,可重复2~4的步骤直到进入系统。

这里应当指出,系统对口令输入的等待时间进行了限制。如果在规定的时间内未输入口令,屏幕上会显示出以下信息:

Error reading command input

Timeout period expired

此时,用户按回车键即可重新输入用户名和口令。另外,为了安全起见(如防止他人猜试口令企图非法闯入系统),系统一般还会对用户注册时连续失败的准许次数进行限制。因此,用户

注册时应尽可能减少失败的次数。若实属口令遗忘,请让系统管理员帮助处理。

当用户成功进入系统时,屏幕上常会先显示一些信息,其典型信息为:

Welcome to VAX/VMS V5.0

last interactive login on Thursday, 16 - JUL - 1992 08:47

last non-interactive login on Thursday, 9 - JAN - 1992 11:33

\$

紧接着出现DCL 命令提示符\$。于是可输入命令开始工作。

#### 5. 改变口令

在注册成功之后,用户可用DCL 命令SET PASSWORD 来改变口令。因为口令使用久了难免会暴露,容易失密。因此,定期或不定期改变自己的注册口令是十分必要的。

改变口令的过程如下:

\$SET PASSWORD (回车)

Old password: 输入旧口令(回车)

New password: 输入新口令(回车)

Verification: 输入新口令(回车)

例如:

\$SET PASSWORD (回车)

Old password: MAKEWORK (回车)

New password: NEWDONE (回车)

Verification: NEWDONE (回车)

第二次输入新口令,主要是为了对口令进行核实确认。只有当两次输入的新口令完全一致时,系统才认可本次对口令的改变有效。改变口令时,输入的口令串(无论是旧口令还是新口令)均不显示在屏幕上。

口令可用字母A~Z、数字0~9、\$和下划线组成。一个口令的长度可为0~31个字符,但一般常要求口令的长度 $\geq 6$ 个字符。

在改变口令时,若希望由系统生成口令串,则输入以下命令:

\$SET PASSWORD/GENERATE=[n]

其中n 指定所生成的口令串长度为n~n+2 个字符,不指定此值时,缺省为6,即所生成的口令长度为6~8个字符。这条命令使系统生成出5个随机口令串供用户选择使用。

例如:

\$SET PASSWORD/GENERATE=8

Old password: 输入旧口令(回车)

coweushvie

c-o-weush-vie

daxopclov

dax-op-clov

timethpyuba

timeth-pyu-ba

nonslevoy

non-sle-voy

ealkduhox

ealk-du-hox

Choose a password from this list, or press RETURN to get a new list

New password:

这时输入由系统提供的口令串。如果不喜欢这些口令,可按回车键让系统生成再提供5个新的口令串。

有的用户为了增强安全感,可以向系统管理员申请建立两个注册口令。这时,改变第二个口令要使用以下命令:

\$SET PASSWORD/SECONDARY

## 6. 改变命令提示符

为了符合自己的上机操作习惯,用户可使用SET PROMPT命令改变DCL提示符\$。此命令的格式为:

```
$SET PROMPT=string
```

其中string 指定新的提示字符串。

例如,将提示符\$ 改变成A> 的命令如下:

```
$SET PROMPT=A> (回车)
```

```
A>
```

## 2.2 退出系统

用户在系统中完成所做工作想离开机房时,应及时退出系统。为了退出系统,结束终端对话,在\$ 提示符下输入LOGOUT 命令,即

```
$ LOGOUT
```

可缩写成:

```
$ LO
```

输入命令之后,系统立即显示出用户退出系统的日期和时间。例如:

```
USERGAS logged out at 16-JUL-1992 12:14:50.12
```

如果想知道自己从注册到退出这期间用了多少终端时间,消耗了多少CPU 时间以及其它记帐信息,用户在输入的LOGOUT 命令后面加上限定词/FULL 即可。

例如:

```
$ LOGOUT/FULL
```

```
USERGAS          logged out at 16-JUL-1992 09:22:30.30
```

### Accounting information:

|                     |               |                        |               |
|---------------------|---------------|------------------------|---------------|
| Buffered I/O count: | 182           | Peak working set size: | 449           |
| Direct I/O count:   | 41            | Peak page file size:   | 2433          |
| Page faults:        | 671           | Mounted volumes:       | 0             |
| Charged CPU time:   | 0 00:00:02.98 | Elapsed time:          | 0 00:04:45.61 |

一旦退出系统之后,用户若想再次使用系统,必须重新进行注册操作。

## 第三章 终端操作

在VAX 机上所使用的终端类型较多,例如有VT100系列、VT200系列以及VT300系列等终端。要用好终端,应熟悉终端的屏幕和键盘及其操作特性的设置与使用。下面主要以VT320终端介绍之。

### 3.1 屏幕与键盘

VT320是一种通用的视频显示终端,主要由显示屏幕和键盘组成。VT320终端如图3.1所示。

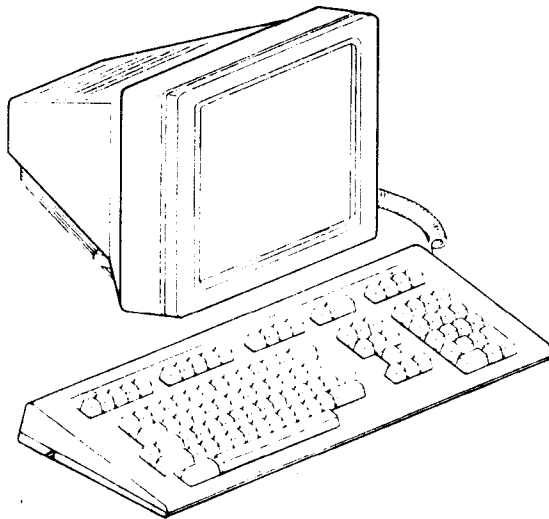


图 3.1

#### 3.1.1 屏幕

VT320 使用的是一种大小为356mm(14英寸)的单色显示屏幕,共有25行,可以显示24行、80列或132列的文本内容。第25行为终端状态行。它可以和主机、终端服务器或调制解调器相连接,还可以连接一台打印机。

VT320 具有一种延长屏幕寿命的CRT Saver 特性。如果在30分钟内无键盘操作或来自主机系统的输入,显示屏幕将自动地变为空白,但并不丢失原先所显示的内容。如果要使显示屏幕重新工作,按任一键即可激活。

#### 3.1.2 键盘

VT320 使用的键盘如图3.2 所示。



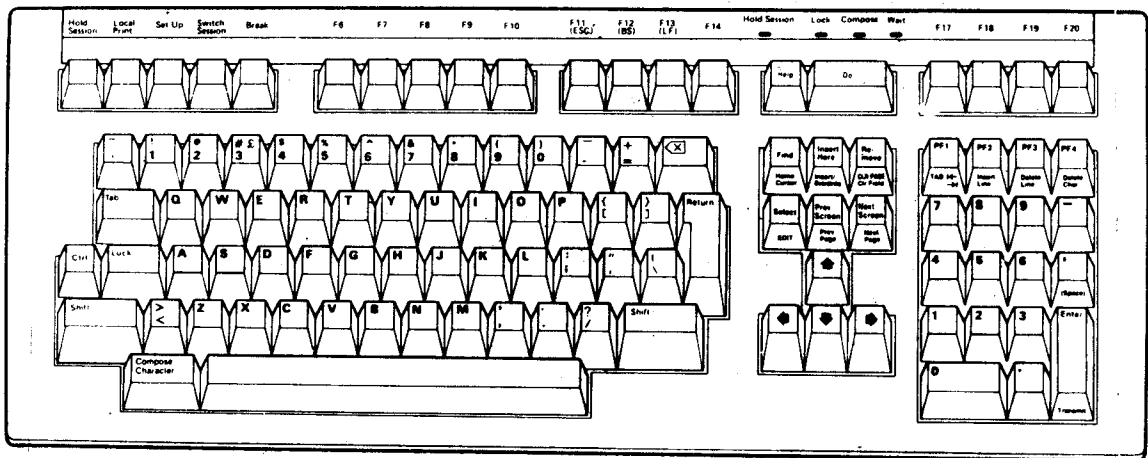


图 3.2

键盘上有以下四组键:

- 主键盘
- 编辑键盘
- 数字键盘
- 顶排功能键

另外还有四个指示灯和两个发声指示器。

#### 1. 主键盘

主键盘如图3.3所示,它与标准的打字机键盘相类似。

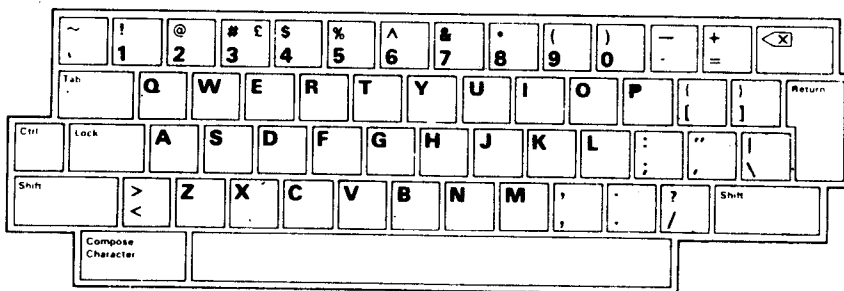


图 3.3

主键盘上有下列特殊功能键:

- Tab** Tab 键为水平制表键,它使光标移动到本行中的下一个Tab 位置。用户可使用设置“Tab Set-Up”来选择Tab 制表位置。
- CTRL** 按住CTRL键,并按下另一任意键,向系统发送一个控制代码。控制代码使系统执行某一特定操作。例如,CTRL/U表示按住CTRL键并按下U键,于是删除本行中从光标到行首之间的所有字符。
- LOCK** 若使用了“Keyboard Set-Up”中的“Caps Lock”设置,按下此键后(此时LOCK指示灯亮),按字母键时所送出的是大写字母。如果使用的是“Shift Lock”设置,当按下LOCK键时,以后所按键送出的是键上的顶部字符;当释放