

IBM PC/AT

硬件和 XENIX 系统资料汇编之(三)

XENIX 命令参考手册

曾昱满	周晓萱	译
王会平	张亦明	
孙玉方		校

北京科海总公司培训中心
中国科学院软件研究所

一九八七年八月

IBM PC/AT 硬件和 XENIX 系统资料汇编

(之三)

XENIX 命令参考手册

曾昱满	周晓萱	译
王会平	张亦明	
孙玉方		
		校

北京科海总公司培训中心
中国科学院软件研究所

一九八七年八月

编辑：科海培训中心教材部
发行：科海培训中心资料组
地址：北京2725信箱 科海培训中心
资料组

(北京海淀路黄庄文化馆大门右侧平房)

印刷：河北省蔚县印刷厂

编 者 序

IBM PC已从PC、PC/XT推进到PC/AT。PC/AT以Intel 80286为主CPU，具有丰富的硬件资源。鉴于目前DOS系统基本上是一种单用户系统，许多硬件资源未得到充分利用，许多用户都要求在PC/AT上配备多用户多任务的XENIX系统。

XENIX系统是UNIX系统在以Intel为主CPU的微机上的实现，该系统由Microsoft公司开发。目前在PC/AT上运行的XENIX相当于UNIX的System III或System V。

为了更好地在国内推广PC/AT及其兼容机，中国科学院软件研究所在其雄厚的技术基础上，积多年研究、开发UNIX系统之经验，开发成功多种XENIX中西文信息处理系统并移植到几乎所有PC/AT的兼容机上。为了更好地推广XENIX中英文信息处理系统，科海培训中心和中国科学院软件研究所组织了一批专家和技术人员，收集和编译整理了有关XENIX及IBM PC/AT的全部技术资料。

这些资料包括以下几类：

- I、IBM PC/AT 硬件资料，包括硬件安装及组装手册、技术手册和维护手册。
- II、XENIX基本系统的安装、基本用户指南、命令参考手册、系统管理手册和直观Shell手册。
- III、XENIX开发系统方面的软件开发手册、库函数程序员手册、系统调用和子程序手册。
- IV、XENIX系统上运行的汇编语言和各种高级语言（C、Fortran、Pascal、Cobol、Basic）的用户指南和参考手册。
- V、XENIX正文格式处理手册。
- VI、XENIX系统上配备的最新版本INFORMIX数据库管理系统用户参考手册。
- XII、中西文兼容的C—XENIX系统安装和基本使用参考手册。

全套资料约400万字，分装成20本。

全书的主要译校任务由中国科学院软件研究所的专家、技术人员承担，科海培训中心负责编辑、印刷和发行工作。

由于时间仓促，本资料汇编中必有不少错漏之处，敬请读者批评指正，以便再版时更正。

主 编 孙玉方
 董洪泉

本书概述

本手册是提供给使用 XENIX 操作系统的程序员的参考资料。本手册列出了 XENIX 操作系统的所有 (C)、(M) 和 (F) 节的命令。(C) 为命令, (M) 为各种杂项信息, (F) 为文件格式。

这本书里给出了各个命令的名字、句法、说明和例子, 必要时加上注释。(M) 节中有各种杂项信息, 其中包括大量的系统维护信息。(F) 节罗列了各种文件的格式。

与本手册有关的 XENIX 资料包括:

- IBM PC XENIX 安装指南;
- IBM PC XENIX 直观Shell;
- IBM PC XENIX 系统管理手册;
- IBM PC 基本操作指南。

目 录

第一部分 XENIX 操作系统命令.....	(1)
命令 (C) 概述.....	(1)
ACCTCOM (C)	(1)
ACCTON (C)	(3)
ASKTIME (C)	(3)
ASSIGN (C)	(4)
AT (C)	(5)
AWK (C)	(6)
BACKUP (C)	(9)
BANNER (C)	(10)
BASENAME (C)	(10)
BC (C)	(11)
BDIFF (C)	(13)
BFS (C)	(14)
CAL (C)	(17)
CALENDAR (C)	(17)
CAT (C)	(18)
CD (C)	(18)
CHGRP (C)	(19)
CHMOD (C)	(19)
CHOWN (C)	(21)
CHROOT (C)	(21)
CHSH (C)	(22)
CMP (C)	(22)
COMM (C)	(22)
COPY (C)	(23)
CP (C)	(24)
CPIO (C)	(24)
CRON (C)	(25)
CSPLIT (C)	(26)
CU (C)	(27)
DATE (C)	(29)
DC (C)	(31)
DD (C)	(32)
DEVNM (C)	(33)

DF (C)	(34)
DIFF (C)	(34)
DIFF 3 (C)	(35)
DIRCMP (C)	(36)
DIRNAME (C)	(36)
DISABLE (C)	(37)
DOS (C)	(38)
DTYPE (C)	(39)
DU (C)	(40)
DUMPDIR (C)	(40)
ECHO (C)	(41)
ED (C)	(42)
ENABLE (C)	(49)
ENV (C)	(49)
EX (C)	(49)
EXPR (C)	(51)
FACTOR (C)	(52)
FALSE (C)	(53)
FILE (C)	(53)
FIND (C)	(54)
FINGER (C)	(55)
FSCK (C)	(56)
GETOPT (C)	(59)
GREP (C)	(59)
GRPCHECK (C)	(61)
HAITSYS (C)	(61)
HD (C)	(62)
HEAD (C)	(63)
ID (C)	(63)
INSTALL (C)	(63)
JOIN (C)	(64)
KILL (C)	(65)
LC (C)	(65)
LINE (C)	(67)
LN (C)	(67)
LOGNAME (C)	(68)
LOOK (C)	(68)
LIR (C)	(68)
LS (C)	(69)

MAIL (C)	(70)
MESG (C)	(76)
MKDIR (C)	(76)
MKFS (C)	(77)
MKNOD (C)	(78)
MKUSER (C)	(78)
MORE (C)	(79)
MOUNT (C)	(82)
MV (C)	(82)
NCHECK (C)	(83)
NETUTIL (C)	(83)
NEWGRP (C)	(84)
NEWS (C)	(85)
NICE (C)	(85)
NL (C)	(86)
NOHUP (C)	(87)
OD (C)	(87)
PACK (C)	(88)
PASSWD (C)	(89)
PR (C)	(90)
PS (C)	(91)
PSTAT (C)	(94)
PWADMIN (C)	(96)
PWCHECK (C)	(97)
PWD (C)	(97)
QUOT (C)	(98)
RANDOM (C)	(98)
RCP (C)	(99)
REMOTE (C)	(100)
RESTORE (C)	(100)
RM (C)	(101)
RMDIR (C)	(102)
RMUSER (C)	(102)
RSH (C)	(103)
SDDATE (C)	(104)
SDIFF (C)	(104)
SED (C)	(105)
SETMNT (C)	(107)
SETTIME (C)	(108)

SH (C)	(108)
SHUTDOWN (C)	(115)
SLEEP (C)	(116)
SORT (C)	(116)
SPLIT (C)	(117)
STTY(C)	(117)
SU (C)	(120)
SUM (C)	(121)
SYNC (C)	(121)
SYSADMIN(C)	(121)
TAIL(C)	(122)
TAR (C)	(122)
TEE(C)	(125)
TEST(C)	(125)
TOUCH (C)	(127)
TR (C)	(127)
TRUE(C)	(128)
TSET (C)	(128)
TTY (C)	(130)
UMASK (C)	(130)
UMOUNT (C)	(130)
UNAME (C)	(131)
UNIQ (C)	(131)
UNITS (C)	(132)
UUCLEAN (C)	(133)
UUCP (C)	(133)
UUSTAT (C)	(135)
UUSUB (C)	(136)
UUTO (C)	(137)
UUX (C)	(138)
VI (C)	(139)
VSH (C)	(166)
WAIT (C)	(166)
WALL (C)	(166)
WC (C)	(167)
WHAT (C)	(167)
WHO (C)	(167)
WHODO (C)	(168)
WRITE (C)	(168)

XARGS (C)	(169)
YES (C)	(170)
第二部分 维护命令和其它信息	(172)
引言	(172)
ALIASES (M)	(172)
ALIASHASH (M)	(173)
ASCII (M)	(174)
BADTRACK (M)	(175)
BOOT (M)	(176)
CLOCK (M)	(177)
CMOS (M)	(178)
CONSOLE (M)	(178)
DAEMON.MN (M)	(180)
DEFAULT (M)	(180)
ENVIRON (M)	(181)
FD (M)	(182)
FDISK (M)	(184)
FORMAT (M)	(185)
GETTY (M)	(185)
GROUP (M)	(186)
HD (M)	(186)
INIT (M)	(187)
LD (M)	(188)
LOGIN (M)	(190)
LP (M)	(192)
MEM (M)	(192)
MESSAGES (M)	(192)
MICNET (M)	(196)
NULL (M)	(197)
PASSWD (M)	(197)
PROFILE (M)	(197)
SERIAL (M)	(198)
SETCLOCK (M)	(199)
SETKEY (M)	(199)
SYSTEMID (M)	(199)
TERM (M)	(200)
TERMCAP (M)	(201)
TERMINALS (M)	(209)
TOP (M)	(210)

TTY (M)	(211)
TTYS (M)	(218)
UTMP (M)	(218)
XINSTALL (M)	(219)
第三部分 文件格式.....	(221)
文件格式介绍.....	(221)
A.OUT (F)	(221)
ACCT (F)	(221)
AR (F)	(222)
BACKUP (F)	(222)
CHECKLIST (F)	(223)
CORE (F)	(223)
CPIO (F)	(223)
DIR (F)	(224)
FILESYSTEM (F)	(225)
INODE (F)	(226)
MASTER (F)	(226)
MNTTAB (F)	(228)
SCCSFILE (F)	(228)
STAT (F)	(230)
TYPES (F)	(231)

第一章 XENIX操作系统命令

命令(C)概述

intro-XENIX命令概述。

说明

本节说明了XENIX操作系统中每个可用命令的使用方法。

句法

除特别说明外，本节中说明的命令接受按下列句法构成的任选项和自变量。

name [options] [cmdargs]

其中：

name

是可执行文件名，必须被准确输入。

options

-noargletters 或 -argletter< >optarg。其中< >是任意多个空格。

noargletters

一个单个字母，表示一个不带自变量的任选项。

argletter

一个单个字母，表示一个要求带自变量的任选项。

optarg

满足其前面 argletter 的自变量（字符串）。

cmdarg

不以减号（-）开头的路径名（或其它命令自变量）。减号（-）表示标准输入，

参见

getopt (C) , getopt (S)

诊断

每条命令结束都返回两个字节的状况信息。其中一个是由系统提供的，它给出了命令结束的原因；另一个（在“正常”结束时）是由程序提供的。前一字节为0表示正常结束；后一字节通常以0表示命令执行成功，如果不为0，则表示出了问题，比如错误的参数，不好的或者不可访问的数据，这后一位有几种称呼：‘出口码’（exit code），‘出口状态’（exit status）及‘返回码’（return code）。它仅在有特别约定时才给予解释。

注释

并非所有命令都要求任选项和自变量。

ACCTCOM (C)

名字

acctcom—查找并打印进程记帐文件。

句法

acctcom [[options] [file]] ...

说明

acctcom 命令按照 acct(F) 描述的格式读 file、标准输入、或/usr/adm/pacct 文件，并将所选记录写到标准输出上。每一条记录代表一个程序的执行。输出显示 COMMAND、USER、TTYNAME、START TIME、ENDTIME、REAL(SEC)、CPU(SEC)、MEAN SIZE(K) 以及可有可无的 F(fork/exec 标志：1 表示不带 exec 的 fork) 和 STAT (系统退出状态)。

如果该命令是由具有特权的超级用户执行的，则其命令名带前缀“.”；如果进程未同 一个可识别的终端相关，则在 TTYNAME 列打印一个“?”。

如果没有指出 file 并且标准输入与一个终端或/dev/null (就如 shell 中使用“&”的情况) 相关，则本命令读/usr/adm/pacct 文件，否则读标准输入文件。

如果给出自变量 file，则按其各自的顺序读文件。每一个文件通常向前读，即按各进程完成的时间顺序读。文件/usr/adm/pacct 通常是当前被检测的文件，对于一个较繁忙的系统需若干个文件，在这种情况下，除当前文件外，所有文件均可在 /usr/adm/pacct? 中找到。

任选项有：

- b 向后读，最后一个命令最先显示。
- f 在输出中打印 fork/exec 标志和系统退出状态列。
- h 不显示平均内存长度，而显示该进程执行期间消耗的 CPU 时间百分比。这一“变形因数”如下计算：(总 CPU 时间/消耗的时间)。
- i 在输出中打印 I/O 记数。
- k 不显示内存长度而显示总核心占有的时间。
- inv 显示平均核心长度 (缺省)。
- r 显示 CPU 因数 (用户时间/(系统时间 + 用户时间))。
- t 分别显示系统和用户所占 CPU 时间。
- u 去掉输出中的标题。
- l line 仅显示终端/dev/line 上的程序。
- u user 仅显示属于可由用户 ID 指定的 user 的进程，可转换为用户 ID 的注册名，一个“.” (它指定仅是带有超级用户特权执行的进程) 或“?” (它指定仅仅是那些与未知的用户 ID 相关的进程)。
- g group 仅显示属于组 group 的进程。这个组可由组 ID 或组名指定。
- d mm/dd 跟在此任选标志之后的所有的 time 自变量均假定为发生在给定的月份和日期的24小时之内。这对查寻老文件是必需的。
- s time 仅显示存在的进程或以 hr:min:sec (时:分:秒) 形式给出的 time 之后的进程。:sec 或:min:sec 可省略。
- e time 仅显示存在的或在 time 之前的进程。对于-s 和-e 使用同一个 time 显示在 time 时存在的进程。
- n pattern 仅显示与 pattern 匹配的命令。除了“+”表示出现一次或多次以外，pattern 是一个同 ed (C) 中一样的正则表达式。

- H factor 仅显示超过 factor 的进程。factor 是在-h任选项中解释的“变形因数”。
- O time 仅显示其占用操作系统 CPU 时间超过 time 的进程。
- C time 仅显示超过 time (总 CPU 时间) 的进程。

复合任选项为逻辑 AND 关系。

文件

/etc/passwd
/usr/adm/pacct
/etc/group

参见

accton(C) , ps(C) , su(C) , acct(F) , utmp(M)

注释

acctcom命令仅报告已结束的进程；用 ps(C) 可看活动进程。

“11:29 April 20, 1984”

ACCTON(C)

名字

accton - 启动记帐。

句法

accton [file]

说明

accton命令启动和停止记帐进程。如果未给出file, 则停止记帐；如果给出了file, 则它必须是一个已存在的文件名, 核心将记帐记录附加在该文件中(见acct(F))

文件

/etc/passwd 用于注册名和用户ID的转换。
/usr/tib/acct 包含许多记帐命令。
/usr/adm/pacct 当前进程记帐文件。
/usr/adm/wtmp 注册/注销历史文件。

参见

acctcom(C), acct(F), utmp(M)

ASKTIME (C)

名字

asktime—提示要求当日的正确时间。

句法

/etc/asktime

说明

该命令提示要求当日时间。必须按下面定义的格式输入一个合法时间：

[yyymmdd] hhmm

这里yy是年的最后两位数；前一个mm是月份；dd是月中的日期。日期可有可无。如果你不输入日期, 则用当前的年、月、日 hh是小时(24小时制), 第二个mm是分钟。

例子

本例设置新的时间、日期和年为：“11:29 April 20, 1984”

当前系统时间是: Wed Nov 3 14:36:23 PST 1982。

输入时间 ([yyymmdd] hhmm) : 8404201129

诊断

如果你输入非法时间, 则 asktime 显示:

```
cvtdat: bad conversion
```

并退出。

注释

asktime 命令通常在系统引导之后立即由系统配置文件/etc/rc 自动完成, 但它可在任何时候执行。该命令有特权限制, 只能由超级用户执行。

ASSIGN (C)

名字

assign, deassign—分配和取消设备。

句法

```
assign [-u] [-v] [-d] [device] ...
```

```
deassign [-u] [-v] [device] ...
```

说明

assign 命令试图为当前用户分配 device。变量 device 必须是一个当前未被分配的可分配设备。不带自变量的 assign 命令按照被指定用户的名字打印一组可分配的设备。

deassign 命令用于取消设备。不带自变量的 deassign 命令将取消所有分配给用户的设备。若给出自变量, 则试图取消作为自变量给定的 device。

任选项有:

-d 执行 deassign 操作。

-v 给出详细的输出。

-u 不进行“分配”和“取消”, 但执行错误检查。

assign 命令如果不能分配所有可分配设备, 则它就不分配任何设备。deassign 命令在 device 不可取消时不进行诊断。设备可在退出注册时自动取消, 但不能保证取消。device 名可能是最初要求的设备。例如: assign fd 通常能分配所有盘设备。device 的原始版本也可被分配。例如原始盘设备/dev/rfd? 在上例中也将被分配。

文件

/etc/atab	可分配的设备表
/dev/asglock	防止并发存取文件

诊断

若成功返回则退出码 0, 有问题时返回 1, 若 device 不能打开则返回 2。

注释

在许多装置中, 象盘驱动器那样的可分配设备通常可读写存取, 所以 assign 命令可能是不必要的。在单用户系统中, 尤其如此。用此命令分配的设备应属于用户, asg 目录 /dev 应属于/bin 且具有 755 存取方式。assign 命令 (在检查完使用者之后) 无论谁使用都只配设备而不改变存取权限。允许系统管理员设置各个设备的自由使用权、可分配权 (属于 asg)、不可分配权和限制权限 (不属于 asg 且有某些限制方式)。

首次调用 `assign` 建立可分配设备表 `/etc/atab`，这个表格用于随后的访问以节省对 `/dev` 目录的重复查找。如果 `/dev` 中某一设备被改为可分配的（即 `asg` 拥有的设备），则超级用户必须删除 `/etc/atab` 表，使得在下一次命令使用时能建立一个正确的表格。

AT (C)

名字

`at`、`atg`、`atrm`—在以后的某一时刻执行命令。

句法

```
at time [day]
    [file]
atg [-l]
atrm idnumber...
```

说明

`at` 命令导致文件的内容在指定的时刻作为一个 `shell` 文件执行，此命令对于在固定时刻或在系统空闲时，运行进程是非常有用的。

自变量有：

time 1 ~ 4 位数字，后跟一任选项 “a” 表示 am，“p” 表示 pm，“n” 表示中午，“m” 表示午夜。1 ~ 2 位数字解释为小时；3 ~ 4 位数字则解释为时和分。若在数字后面不跟字母，则假定为 24 小时制。

day 可以是后跟天数的月份名或一星期中的某天。如果后面接一个字 “week”，则要在所指定的日期再加上 7 天时间后引用文件。月份名和日期都能按公认的方式缩写（见本节后面的例子）。

file 文件名包含被执行的命令。若是没有指定 `file`，则认为是标准输入。

`at` 命令建立一个在指定时间由 `shell` 执行的文件，该文件包含：用户 ID 及用户组 ID 的注释行；一个 `cd` 命令——用于在执行 `at` 时将工作目录转到你要使用的目录上；为环境变量赋值及在 `at` 命令上指定的 `file`。文件中的输出必须改向，否则将被丢失。`at` 命令的 `shell` 文本将由 `cron` (C) 定期执行 `/usr/lib/atrun` 来运行。

`atq` 命令给出等待处理的文件的下列信息：

- 将运行文件的用户 ID。
- 用于代表文件的唯一标识 (ID) 号。
- 文件执行的日期和时间。
- 任选项显示队列中每一个文件中的命令。

`atrm` 命令从 `at` 队列中删除文件。`atrm` 命令根据 `atq` 命令中的 ID 号删除指定的文件，一个用户只能删除其自己的文件。

例子

用下列命令将文件放入队列：

```
at 8a jan 24 file
```

在下面的命令行中，`file` 将从本星期五下午 3 : 30 起执行一个星期：

```
at 1530 fr week file
```

要删除队列中的一个文件，可用 `atq` 找到 ID 号，然后用 `atrm` 删除该文件：

atrm idnumber

文件

/usr/spool/at/vy.ddd.hhhh.uu

在yy年ddd日hhhh时完成的活动，uu是一单独的数

/usr/spool/at/lasttimedone

记有活动的最后小时hhhh。

/usr/spool/at/past

包含老的 at 文件。

/usr/lib/atrun

在指定时刻执行活动的程序。

参见

calendar(C), cron(C), pwd(C)

诊断

报告各种句法错误及超出规定范围的时间。

注释

目录/usr/spool/at/past 应定期由超级用户清空。

由于执行/usr/lib/atrun 的粒度，在今后24小时内的调度方面也许会有问题。

AWK (C)

名字

awk——查找和处理文件中的模式。

句法

awk [-Fc] [-f programfile +]'program' [file...]

说明

awk命令扫描每一个输入文件 file，以寻找匹配 program 和 programfile 中指定的模式的行。当 file 中某一行与模式相匹配时就执行对应的动作。该命令对于编译信息，完成输入数据中的算术运算以及进行重复或条件处理是非常有用的。

任选项有：

-Fc 设置字段分隔符为“c”，缺省时字段分隔符为制表符或空白。

-f 让 awk 从 programfile 中取程序。

自变量有：

programfile 包含 awk 程序的文件。

program awk程序。程序所在命令行必须由单引号括起来以防 shell 解释。

file... 被处理的文件名。若未给出文件名，则使用标准输入。

awk 程序由下列形式的语句组成：

pattern { action }

模式-动作语句可出现在 awk 命令行中，也可在 awk 程序文件中。

如果未给出 pattern，则认为输入文件中所有行均匹配；如果未给出 action，则每个被匹配的行均在标准输出上显示。

模式可以是一个文字串，一个正则表达式，或是一个正则表达式与由运算符分隔的字段