

适用于 IBM PC AT 286 386 及其兼容机

SCOXENIX 系统V

技术丛书

(共五册)

开发系统用户指南和程序员指南



上海电子计算机厂

北京希望电脑公司

前言

XENIX 多用户分时操作系统是 UNIX 系统的派生版本。它适用于 INTEL 公司的 8086/80286/80386 等微机系统。在国际上有广泛的应用,已销售 20 万份以上。最初, XENIX 由 UNIX 版本 7 派生而出,后来出现的 XENIX 系统 III 型和 XENIX 系统 V。近年来,国内的 0530 (286)和 0540(386)微机上也较广泛使用 XENIX 系统 V,如银行、保险公司和高等学校等。XENIX 系统 V 2.2.1 版是 SCO 公司 1987 年 4 月推出的较新版本。它弥补了 XENIX 系统 III 的许多不足之处,如 XENIX 系统 III 只能接 2 型硬盘,而系统 V 可接多种类型硬盘,其容量可以选择。XENIX 系统 V 2.2.1 的主要特点有:

1. 符合 UNIX 系统 V 接口规范 SVID,对共享数据与信号灯、分配内存操作提供两套函数库,分别与 UNIX 标准及 XENIX 以前版本兼容。

2. 具有 UNIX 系统 V 的图形功能,提供对主控适配器口地址和显示缓冲区的操作。对主控监视器有计算机图形接口。

3. 操作系统分成多个软件包,用户可有选择地安装、删除、检查某个包及其中的文件。以达到按用户需要来定制操作系统,并保证操作系统的完整性。

4. 友好的用户界面,系统安装过程用菜单驱动,自动安装。引导过程灵活,同一核心既可引导软盘;还可引导 DOS 操作系统。主控监视器有多屏功能。外设驱动程序设计灵活,可接较多的常用外部设备。

自 1987 年 10 月以来我们网络组开发了两种汉字 XENIX 系统 V 2.2.1 操作系统。第一种主控监视工作于汉字字符方式,可采用长城 014 卡、CEGA 卡、东海单色汉卡、先锋汉卡。第二种主控监视器工作于图形方式。用内存 16×16 点阵字库(俗称软字库)、可采用 SEGA 卡(640×480)、扩展 EGA 卡(640×350)、COLOR-400 卡(640×400)。此外,在长城 286 上研制了以 TCP/IP 协议为主体的 XENIX 系统 V 环境下的多用户网络操作系统 PUNET2.0。这些软件产品于 1988 年第一季度试用改进将从 6 月起陆续由有关公司、工厂向用户提供。

为了对 XENIX 系统 V 的用户提供方便,应有关公司和工厂要求,我们组织了 XENIX 系统 V 2.2.1 全套说明书(共七卷)的翻译工作。因参加者对 XENIX 系统熟悉程度和英文水平的不同,加上时间短,交稿急,错译、误校、有的地方语言不解顺畅,排印也有某些欠妥之处在所难免。望读者批评指正,待重印时修正。

这套说明书共争七卷:

XENIX 系统 V 操作系统

第一卷 XENIX 系统 V 运行环境

第一部分:安装指南 李红 译 杨雄 校

第二部分: XENIX 系统 V 2.2.1 版本注释 郭宏 译 余娟芬 校

第三部分: XENIX 入门 周翔 译 余娟芬 校

第四部分:操作指南 杨志慧 胡文绮 译 冯运华 校

第五部分:硬件相关部分 郭宏 译 余娟芬 校

第二卷 XENIX 系统 V 用户参考手册

第一部分:命令参考手册

张永明 陈凡 朱礼方 译 郭宏 余娟芬 朱征南 校

第二部分:杂项命令及信息 李红 译 丛远华 校

第三部分:文件格式 朱征南 译 杨雄 校

第三卷 用户指南

程竹章 朱征南 朱礼方 周欣 译

胡文绮 周炜 余娟芬 李红 校

XENIX 系统 V 开发系统

第四卷:程序设计命令

第一部分:程序设计命令 钟维德 朱征南 译 余娟芬 校

第二部分:系统服务

朱征南 李敏 程竹章 李红 译

余娟芬 葛衡 李红 校

第三部分:DOS 例程 熊志国 译 朱征南 校

第五卷:程序员指南(上)

第一部分:开发系统版本说明 熊志国 译 朱征南 校

第二部分:程序员指南 熊志国 译 朱征南 校

第三部分:C 语言参考手册 林杨 译 丛远华 校

第四部分:宏汇编用户指南 张永明 译 余娟芬 校

第五部分:宏汇编参考手册 张永明 译 余娟芬 校

第六卷:程序员指南(下)

第一部分:C 用户指南

朱征南 熊志国 葛衡 译 郭宏 李红 校

第二部分:C 库指南 葛衡 译 郭宏 校

XENIX 系统 V 正文排版系统

第七卷:XENIX 正文排版系统

第一部分:用户指南 黄瑛 译 余娟芬 校

第二部分:用户参考手册

王晓 黄瑛 王凯 周永军 译

葛衡 郭宏 余娟芬 校

这套资料由北京大学新技术公司电脑激光照排中心用华光 IV 轻印刷系统编辑排版印刷。

字母顺序表命令,系统调用,库例程及文件格式

a.out	a.out(F)	awk	awk(C)
a641	a641(S)	backup	backup(C)
abort	abort(S)	backup	backup(F)
abs	abs(S)	badtrk	badtrk(M)
accept	accept(C)	banner	banner(C)
access	access(S)	basename	basename(C)
acct	acct(F)	batch	at(C)
acct	acct(S)	bc	bc(C)
accteom	accteom(C)	bdiff	bdiff(C)
accton	accton(C)	bdos	bdos(DOS)
acos	trig(S)	bessel	bessel(S)
abd	abd(CP)	bfs	bfs(C)
admin	admin(CP)	boot	boot(HW)
alarm	alarm(S)	brk	brk(S)
aliases	aliases(M)	brkctl	brkctl(S)
aliases.hash	aliases(M)	bsearch	bsearch(S)
aliashash	aliashash(M)	cabs	hypot(S)
ar	ar(CP)	cal	cal(C)
ar	ar(F)	calendar	calendar(C)
ascii	ascii(M)	calloc	malloc(S)
asctime	ctime(S)	cancel	lp(C)
asin	trig(S)	cat	cat(C)
asktime	asktime(C)	cb	cb(CP)
assert	assert(S)	cc	cc(CP)
assign	assign(C)	cd	cd(C)
asx	asz(CP)	cdc	cac(CP)
at	at(C)	ceil	floor(S)
atan	trig(S)	cflow	cflow(CP)
atan2	trig(S)	cgets	cgets(DOS)
atof	atof(S)	chdir	chdir(S)
atof	strtol(S)	checkcw	cap(CT)
atoi	atof(S)	checkeq	eqn(CT)
atoi	atrtol(S)	checklist	checklist(F)
atol	atof(S)	checkmm	checkmm(CT)
atol	atrtol(S)	chgrp	chgrp(C)
autoboot	autoboot(M)	chmod	chmod(C)

chmod	<i>chmod</i> (S)	ctags	<i>ctags</i> (CP)
chown	<i>chown</i> (C)	ctermid	<i>ctermid</i> (S)
chown	<i>chown</i> (S)	ctef	<i>ctef</i> (CP)
chroot	<i>chroot</i> (C)	cron	<i>cron</i> (C)
chroot	<i>chroot</i> (C)	cscanf	<i>cscanf</i> (DOS)
chsize	<i>chsize</i> (S)	csd	<i>csd</i> (C)
clearerr	<i>ferror</i> (S)	csplit	<i>csplit</i> (C)
clock	<i>clock</i> (M)	ctags	<i>ctags</i> (CP)
clock	<i>clock</i> (S)	ctermid	<i>ctermid</i> (S)
clockrate	<i>clockrate</i> (C)	ctime	<i>ctime</i> (S)
close	<i>close</i> (S)	ctype	<i>ctype</i> (S)
closedir	<i>directory</i> (S)	cu	<i>cu</i> (C)
clri	<i>clri</i> (C)	curses	<i>curses</i> (S)
cmchk	<i>cmchk</i> (C)	cuserid	<i>cuserid</i> (S)
cmos	<i>cmos</i> (HW)	custom	<i>custom</i> (C)
cmp	<i>cmp</i> (C)	cut	<i>cut</i> (CT)
col	<i>col</i> (CT)	cw	<i>cw</i> (CT)
comb	<i>comb</i> (CP)	cwcheck	<i>cw</i> (CT)
comm	<i>comm</i> (C)	cxref	<i>cxref</i> (CP)
config	<i>config</i> (CP)	daemon.mn	<i>daemon.mn</i> (M)
console	<i>console</i> (HW)	date	<i>date</i> (C)
conv	<i>conv</i> (S)	dbm	<i>dbm</i> (S)
copy	<i>copy</i> (C)	dc	<i>dc</i> (C)
core	<i>core</i> (F)	dd	<i>dd</i> (C)
cos	<i>trig</i> (S)	deassign	<i>assign</i> (C)
csd	<i>csd</i> (S)	default	<i>default</i> (M)
cp	<i>cp</i> (C)	defopen	<i>defopen</i> (S)
cpio	<i>cpio</i> (C)	defread	<i>defopen</i> (S)
cpio	<i>cpio</i> (F)	delete	<i>dbm</i> (S)
cpr	<i>cpr</i> (CP)	delta	<i>delta</i> (CP)
cprintf	<i>cprintf</i> (DOS)	deroff	<i>deroff</i> (CT)
cputs	<i>cputs</i> (DOS)	devnm	<i>devnm</i> (C)
creat	<i>creat</i> (S)	df	<i>df</i> (C)
creatsem	<i>creatsem</i> (S)	dial	<i>dial</i> (M)
cref	<i>cref</i> (CP)	dial	<i>dial</i> (S)
cron	<i>cron</i> (C)	diction	<i>diction</i> (CT)
cscanf	<i>cscanf</i> (DOS)	diff	<i>diff</i> (CT)
csd	<i>csd</i> (C)	diff3	<i>diff3</i> (C)
csplit	<i>csplit</i> (C)	diffmk	<i>diffmk</i> (CT)

dir	<i>dir</i> (F)	edata	<i>end</i> (S)
dircmp	<i>dircmp</i> (C)	egrep	<i>grep</i> (C)
directory	<i>directory</i> (S)	enable	<i>enable</i> (C)
dirname	<i>dirname</i> (C)	end	<i>end</i> (S)
disable	<i>disable</i> (C)	endgrent	<i>getgrent</i> (S)
diskcmp	<i>diskcp</i> (C)	endpwent	<i>getpwent</i> (S)
diskcp	<i>diskcp</i> (C)	enduntent	<i>getut</i> (S)
divvy	<i>divvy</i> (C)	env	<i>env</i> (C)
dmesg	<i>dmesg</i> (C)	environ	<i>environ</i> (M)
dos	<i>dos</i> (C)	dof	<i>fof</i> (DOS)
doscat	<i>dos</i> (C)	eqn	<i>eqn</i> (CT)
doscat	<i>dos</i> (C)	eqnchar	<i>eqnchar</i> (CT)
doscsp	<i>dos</i> (C)	eqncheck	<i>eqn</i> (CT)
doscsp	<i>dos</i> (C)	erand48	<i>drand48</i> (S)
dosdir	<i>dos</i> (C)	erf	<i>erf</i> (S)
dosdir	<i>dos</i> (C)	erfc	<i>erf</i> (S)
dosexterr	<i>dosexterr</i> (DOS)	errno	<i>perro</i> (S)
dosld	<i>dosld</i> (CP)	etext	<i>end</i> (S)
dosls	<i>dos</i> (C)	ex	<i>ex</i> (C)
dosls	<i>dos</i> (C)	exec	<i>exec</i> (S)
dosmkdir	<i>dos</i> (C)	execl	<i>exec</i> (S)
dosrm	<i>dos</i> (C)	execle	<i>exec</i> (S)
dosrm	<i>dos</i> (C)	execlp	<i>exec</i> (S)
dosrmdir	<i>dos</i> (C)	execv	<i>exec</i> (S)
dosrmdir	<i>dos</i> (C)	execve	<i>exec</i> (S)
drand48	<i>drand48</i> (S)	execvp	<i>exec</i> (S)
dtype	<i>dtype</i> (C)	—exit	<i>exit</i> (S)
du	<i>du</i> (C)	exit	<i>exit</i> (DOS)
dump	<i>dump</i> (C)	exit	<i>exit</i> (S)
dump	<i>dump</i> (F)	exp	<i>exp</i> (S)
dumpdir	<i>dumpdir</i> (C)	explain	<i>explain</i> (CT)
dup	<i>dup</i> (S)	expr	<i>exp</i> (C)
dup2	<i>dup</i> (S)	fabs	<i>floor</i> (S)
8087	<i>8087</i> (HW)	factor	<i>factor</i> (C)
8087	<i>8087</i> (M)	faliases	<i>aliases</i> (M)
86rel	<i>86rel</i> (F)	false	<i>false</i> (C)
echo	<i>echo</i> (C)	fclose	<i>fclose</i> (DOS)
ecvt	<i>exit</i> (S)	fclose	<i>fclose</i> (S)
ed	<i>ed</i> (C)	fcloseall	<i>fcloseall</i> (DOS)

fcntl	<i>fcntl</i> (S)	fscanf	<i>fscanf</i> (S)
fcvt	<i>ecvt</i> (S)	fsck	<i>fsck</i> (C)
fd	<i>fd</i> (HW)	fseek	<i>fseek</i> (S)
fdisk	<i>fdisk</i> (C)	fstat	<i>stat</i> (S)
fdopen	<i>fopen</i> (S)	ftell	<i>fseek</i> (S)
feof	<i>ferror</i> (S)	ftime	<i>time</i> (S)
feof	<i>ferror</i> (S)	ftok	<i>stdipc</i> (S)
ferror	<i>ferror</i> (S)	ftw	<i>ftw</i> (S)
fetch	<i>dbm</i> (S)	fwrite	<i>fread</i> (S)
[I]ush	<i>fclose</i> (S)	fxlist	<i>xlist</i> (S)
fgetc	<i>fgetc</i> (DOS)	gamma	<i>gamma</i> (S)
fgetc	<i>getc</i> (S)	gcvt	<i>ecvt</i> (S)
fgetchar	<i>fgetchar</i> (DOS)	get	<i>get</i> (CP)
fgets	<i>fgts</i> (S)	getc	<i>getc</i> (S)
fgrep	<i>grep</i> (C)	getch	<i>getch</i> (DOS)
file system	<i>file system</i> (F)	getchar	<i>getc</i> (S)
file	<i>file</i> (C)	getcwd	<i>getcwd</i> (S)
filelength	<i>filelength</i> (DOS)	getegid	<i>getuid</i> (S)
fileno	<i>ferror</i> (S)	getenv	<i>genenv</i> (S)
find	<i>find</i> (C)	geteuid	<i>geuid</i> (S)
finger	<i>finger</i> (C)	getgid	<i>getuid</i> (S)
firstkey	<i>dbm</i> (S)	getgrent	<i>getgrent</i> (S)
fixhdr	<i>fixhdr</i> (C)	getgrgid	<i>getgrent</i> (S)
fixperm	<i>fixperm</i> (M)	getgrnam	<i>getgrent</i> (S)
floor	<i>floor</i> (S)	getlogin	<i>getlogin</i> (S)
flushall	<i>flushall</i> (DOS)	getopt	<i>getopt</i> (C)
fmod	<i>floor</i> (S)	getopt	<i>getopt</i> (S)
fopen	<i>fopen</i> (S)	getpass	<i>getpass</i> (S)
fork	<i>fork</i> (S)	getpgrp	<i>getpid</i> (S)
format	<i>format</i> (C)	getpid	<i>getpid</i> (S)
fpoff	<i>fpoff</i> (DOS)	getppid	<i>getpid</i> (S)
fpseg	<i>fpseg</i> (DOS)	getpw	<i>getpw</i> (S)
fprintf	<i>printf</i> (S)	getpwent	<i>getpwent</i> (S)
fputc	<i>fputc</i> (DOS)	getpwnam	<i>getpwent</i> (S)
fputc	<i>putc</i> (S)	getpwuid	<i>getpwent</i> (S)
fread	<i>fread</i> (S)	gets	<i>gets</i> (CP)
free	<i>malloc</i> (S)	gets	<i>gets</i> (S)
freopen	<i>fopen</i> (S)	getty	<i>getty</i> (M)
frexp	<i>frexp</i> (S)	gettydefs	<i>gettydefs</i> (F)

getuid	getuid(S)	intro	intro(M)
getut	getut(S)	intro	intro(S)
getutent	getut(S)	ioctl	ioctl(S)
getutid	getut(S)	ipbs	ipbs(C)
getutline	getut(S)	iperm	iperm(C)
getw	getc(S)	ipcs	ipcs(C)
gmtime	ctime(S)	ips	ips(C)
grep	grep(C)	ips	ips(C)
group	group(M)	isalnum	ctype(S)
grpcheck	grpcheck(C)	isalpha	ctype(S)
gsignal	ssignal(S)	isascii	ctype(S)
haltsys	haltsys(S)	isatty	isatty(DOS)
hashcheck	hashcheck(CT)	isatty	uname(S)
hashmake	spell(CT)	isctrl	ctype(S)
hashmake	spell(CT)	isdigit	ctype(S)
hcreate	hsearch(S)	isgraph	ctype(S)
hd	hd(C)	islower	ctype(S)
hd	hd(HW)	isprint	ctype(S)
hdestroy	hsearch(S)	ispunct	ctype(S)
hdr	hdr(CP)	isspace	ctype(S)
head	head(C)	isupper	ctype(S)
help	help(CP)	isxdigit	ctype(S)
hsearch	hsearch(S)	itoa	itoa(DOS)
hyphen	hyphen(CT)	j0	bessel(S)
hypot	hypot(S)	j1	bessel(S)
id	id(C)	jn	bessel(S)
inprint	inprint(C)	join	join(C)
init	init(M)	rand48	rand48(S)
inode	inode(F)	kbhit	kbhit(DOS)
install	install(M)	keyboard	keyboard(HW)
int86	int86(DOS)	kill	kill(C)
int86x	int86x(DOS)	kill	kill(S)
intdos	intdos(DOS)	kmem	mem(M)
intdosx	intdosx(DOS)	l	l(C)
intro	intro(C)	l3tol	l3tol(S)
intro	intro(CP)	l64a	al64(S)
intro	intro(CT)	labs	labs(DOS)
intro	intro(F)	lc	lc(C)
intro	intro(HW)	lcong48	rand48(S)

ld	ld (CP)	lsearch	lsearch (S)
ld	ld (M)	lseek	lseek (S)
ldexp	frexp (S)	ltoa	ltoa (DOS)
lex	lex (CP)	ltoa3	l3toa (S)
lfind	lsearch (S)	m4	m4 (CP)
line	line (S)	machine	machine (HW)
link	link (S)	mail	mail (C)
lint	lint (CP)	make	make (CP)
ln	ln (C)	aliases	aliases (M)
localtime	ctime (S)	aliases.hash	aliases (M)
lock	lock (S)	mallinfo	malloc (S)
lockf	lockf (S)	malloc	malloc (S)
locking	locking (S)	mallopt	malloc (S)
log	exp (S)	man	man (CT)
log10	exp (S)	mapkey	mapkey (M)
login	login (S)	mapscrn	mapkey (M)
logname	logname (C)	mapstr	mapkey (M)
logname	ylogname (S)	masm	masm (CP)
longjmp	setjmp (S)	master	master (F)
look	look (CT)	matherr	matherr (S)
lorder	lorder (CP)	mem	mem (M)
lp	lp (C)	memccpy	memory (S)
lp	lp (HW)	memchr	memory (S)
lp0	lp (HW)	memcmp	memory (S)
lp0	lp (M)	memcpy	memory (S)
lp1	lp (HW)	memory	ymemory (S)
lp1	lp (M)	memset	memory (S)
lp2	lp (HW)	mesg	mesg (C)
lp2	lp (M)	messages	messages (M)
lpadmin	lpadmin (C)	micnet	micnet (M)
lpinit	lpinit (C)	mkdev	mkdev (C)
lpmove	lpsched (C)	mkdir	mkdir (C)
lpr	lp (C)	mkdir	mkdir (C)
lpr	lpr (C)	mkdir	mkdir (DOS)
lpsched	lpsched (C)	mkfs	mkfs (C)
lpshut	lpsched (C)	mknod	mknod (C)
lpstat	lpstat (C)	mknod	mknod (S)
lrand48	drand48 (S)	mkstr	ymkstr (CP)
ls	ls (C)	mktemp	mktemp (S)

mkuser	mkuser(C)	opensem	opensem(S)
mm	mm(CT)	outp	outp(DOS)
mmcheck	checkmm(CT)	pack	pack(C)
mmt	mmt(CT)	parallel	parallel(HW)
mmttab	mmttab(F)	passwd	passwd(C)
modf	frexp(S)	passwd	passwd(M)
monitor	monitor(S)	paste	paste(CT)
more	more(C)	pause	pause(S)
mount	mount(C)	pcat	pack(C)
mount	mount(S)	pclose	popen(S)
movedata	movedata(DOS)	perror	perror(S)
mrnd48	drand48(S)	pg	pg(C)
msgctl	msgctl(S)	pipe	pipe(S)
msgget	msgget(S)	plock	plock(S)
msgop	msgop(S)	popen	popen(S)
multiscreen	multiscreen(M)	pow	exp(S)
mv	mv(C)	pr	pr(C)
nap	nap(S)	prep	prep(CT)
nbwaitsem	waitsem(S)	printf	printf(S)
ncheck	ncheck(C)	proctl	proctl(S)
neqn	neqn(CT)	prof	prof(CP)
neqn	neqn(CT)	profil	profil(S)
netutil	netutil(C)	profile	profile(M)
newform	newform(C)	prs	prs(CP)
newgrp	newgrp(C)	ps	ps(C)
news	news(C)	pstat	pstat(C)
nextkey	dbm(S)	ptrace	ptrace(S)
nice	nice(C)	ptx	ptx(CT)
nice	nice(S)	putc	putc(S)
nl	nl(C)	putch	putch(DOS)
nlist	nlist(S)	putchar	putchar
nm	nm(CP)	putenv	putenv(S)
nohup	nohup(C)	putpwent	putpwent(S)
nrnd48	drand48(S)	puts	puts(S)
nroff	nroff(CT)	pututline	getut(S)
null	null(M)	putw	putw(S)
od	od(C)	pwadmin	getwdmin(C)
open	open(S)	pwcheck	pwcheck(C)
opendir	directing(S)	pwd	pwd(C)

qsort	qsort(S)	sdgetv	sdgetv(S)
quot	quot(C)	sdiff	sdiff(C)
rand	rand(S)	sdleave	sdenter(S)
random	random(C)	sdleave	sdenter(S)
randlib	randlib(CP)	sed	sed(C)
ratfor	ratfor(CP)	seed48	drand48(S)
rcp	rcp(C)	seekdir	directory(S)
rdchk	rdchk(S)	segread	segread(DOS)
read	read(S)	senctl	senctl(S)
readir	directory(S)	semget	semget(S)
realloc	malloc(S)	semop	semop(S)
red	red(C)	serial	yserial(M)
regcmp.	yregcmp(CP)	setbuf	setbuf(S)
regcmp	regex(S)	setclock	setclock(M)
regex	regex(S)	setcolor	setcolor(C)
regexp	regexp(S)	setgid	setuid(S)
reject	accept(C)	setgrent	getgrent(S)
remote	remote(C)	setjmp	setjmp(S)
rename	rename(DOS)	setkey	setkey(M)
restor	restore(C)	setmnt	setmnt(C)
restore	restore(C)	setmode	setmode(DOS)
rewind	fseek(S)	setpgrp	setpgrp(S)
rewinddir	directory(S)	setpwent	getpwent(S)
rm	rm(C)	settime	settime(C)
rm del	rm del(CP)	setuid	setuid(S)
rmdir	rmdir(C)	setutent	getut(S)
rmdir	rm del(DOS)	setvbuff	getvbuff(S)
rmuser	rmuser(C)	agetl	spull(S)
rah	rah(C)	sh	sh(C)
runbig	runbig(C)	shV	sh(C)
sact	sort(CP)	shacfl	shacfl(S)
sbrk	sbrk(S)	shmget	shmget(S)
scanf	scanf(S)	shmop	shmop(S)
secsdiff	secsdiff(CP)	shutdn	shutdn(S)
secsfile	secsfile(F)	shutdown	shutdown(C)
sdate	sdate(C)	signal	signal(S)
sdenter	sdenter(S)	sigsem	sigsem(S)
sdfree	sdfree(S)	sin	trig(S)
sdget	sdget(S)	sinh	sinh(S)

size	size (CP)	strncpy	string (S)
sleep	sleep (C)	strbrk	string (S)
sleep	sleep (S)	strchr	string (S)
socket	socket (CT)	strrev	strrev (DOS)
open	open (DOS)	strset	strset (DOS)
sort	sort (C)	strspn	string (S)
spawnl	spawnl (DOS)	strtod	strtod (S)
spawnvp	spawnl (DOS)	strtok	string (S)
spell	spell (CT)	strtol	strtol (S)
spellm	spell (CT)	strupr	strupr (DOS)
spline	spline (CP)	stty	stty (C)
split	split (C)	style	style (CT)
sprintf	printf (S)	su	su (C)
spull	spull (S)	sulogin	action (C)
sqr	sqr (S)	sum	sum (C)
rand	rand (S)	swab	swab (S)
rand48	rand48 (S)	sync	sync (C)
scanf	scanf (S)	sync	sync (S)
signal	signal (S)	sys—errlist	error (S)
stackuse	stackuse (CP)	sys—err	error (S)
stat	stat (F)	sysadmin	sysadmin (C)
stat	stat (S)	system	system (S)
stdio	stdio (S)	systemid	systemid (M)
stdipc	stdipc (S)	tail	tail (C)
stime	stime (S)	tan	trig (S)
store	dbm (S)	tanh	sinh (S)
strcat	string (S)	tar	tar (C)
strchr	string (S)		tar (F)
strcmp	string (S)		tbl (CP)
strcpy	string (S)	lete	lsearch (S)
strcpn	string (S)		tee (C)
strdup	string (S)		tell (DOS)
string	string (S)	dir	directory (S)
strings	strings (CP)	tmpnam	tmpnam (S)
strip	strip (CP)	term	term (CT)
strlen	string (S)	term	term (F)
strlen	strlen (DOS)	termcap	termcap (M)
strncat	string (S)	termcap	termcap (S)
strncmp	string (S)	terminals	terminals (M)

termio	termio(M)	ttyslot	ttyslot(S)
test	test(C)	twalk	tesarch(S)
ad	ylsearch(S)	types	types(F)
etemt	termcap(S)	tz	tz(M)
etflag	termcap(S)	tzset	ctime(S)
etnum	termcap(S)	ulinit	ulunit(S)
etstr	termcap(S)	ultoa	ultoa(DOS)
oto	yltermcap(S)	umask	umask(C)
time	time(CP)	umask	umask(S)
ne	time(S)	umount	umount(C)
nes	times(S)	umount	umount(S)
tmpfile	tmpfile(S)	uname	uname(S)
pnam	tmpnam(S)	uname	uname(C)
ascii	conv(S)	unget	unget(CP)
lower	conv(S)	ungetc	ungetc(S)
p	top(M)	ungetch	ungetch(DOS)
p.next	yltop(M)	uniq	uniq(C)
uch	touch(C)	units	units(C)
upper	conv(S)	unlink	unlink(S)
uts	termcap(S)	unpack	pack(C)
.....	tr(C)	ustat	ustat(S)
g	trig(S)	utime	utime(S)
troff	troff(CT)	utmp	utmp(M)
true	yltrue(C)	utmpname	getul(S)
tsearch	tsearch(S)	uuclean	uuclean(C)
tset	tset(C)	uucp	uucp(C)
tsort	tsort(CP)	uuninstall	uuninstall(C)
tty	ty(C)	uulog	uucpy(C)
tty	ty(M)	uuname	uucp(C)
l[A-H]	serial(HW)	uupick	uuto(C)
l[a-h]	serial(HW)	uustat	uustat(C)
l[e-h]	serial(M)	uusub	uusub(C)
2[A-H]	serial(HW)	uuto	uuto(C)
2[a-h]	serial(HW)	uux	uux(C)
2[a-h]	serial(M)	val	val(C)
[02-n]	console(HW)	val	val(CP)
[a-h]	serial(M)	vfprintf	vfprintf(S)
ttyname	ttyname(S)	vi	vi(C)
ttys	ttys(M)	view	vi(C)

vprintf	<i>vprintf</i> (S)	write	<i>write</i> (S)
vsh	<i>vsh</i> (C)	wtmp	<i>wtmp</i> (M)
vsprintf	<i>vsprintf</i> (S)	xargs	<i>xargs</i> (S)
wait	<i>wait</i> (C)	xlist	<i>xlist</i> (S)
wait	<i>wait</i> (S)	xref	<i>xref</i> (CP)
waitsem	<i>waitsem</i> (S)	xstr	<i>xstr</i> (CP)
wall	<i>wall</i> (C)	y0	<i>bessel</i> (S)
wc	<i>wc</i> (C)	y1	<i>bessel</i> (S)
what	<i>what</i> (C)	yacc	<i>yacc</i> (CP)
who	<i>who</i> (C)	yes	<i>yes</i> (C)
whodo	<i>whodo</i> (C)	yn	<i>bessel</i> (S)
write	<i>write</i> (C)		

目 录

程序设计命令(CP)

intro	介绍 XENIX 系统开发命令
adb	调用一个通用的调试器
admin	创建和管理 sccs 文件
ar	维护档案和库
asx	XENIX 8086/186/286 汇编程序
cb	美化 C 程序
cc	调用 c 编译器
cdc	修改一个 sccs 文件 delta 的注释
cflow	生成 c 程序流程图
comb	合成 sccs 的 deltas
config	配置一个 XENIX 系统
cpp	c 语言预处理程序
cref	产生一个交叉引用列表
ctags	建立一个标志文件
cxref	生成 c 程序交叉引用
delta	建立一个 sccs 文件的 delta(更改)
dosld	XENIX 到 MS-DOS 交叉链接
get	取一个 sccs 文件版本
gets	从标准输入读取一个串
hdr	显示目标文件的若干部分
help	有关 sccs 命令的求助命令
ld	调用链接编译器
lex	生成词法分析程序
lint	检查 c 语言用法和语法
lorder	找出一个目标库中的顺序关系
m4	调用一个宏处理程序
make	维护、更新和重新生成成组的程序
masm	调用 XENIX 汇编器
m:str	由 c 源文件建立错误消息文件
nm	打印名字表
prof	显示剖面数据
prc	打印一个 sccs 文件
ranlib	将档案库转换成随机库
ratfor	将合理的 FORTRAN 转换成标准的 FORTRAN
rehcmp	编译正则表达式
rmel	从一个 sccs 文件中删除一个 delta

sact	打印当前 scs 文件的编译活动情况
scsdiff	比较一个 scs 文件的两个版本
size	打印目标文件的大小
spline	插值以平滑曲线
stackuse	确定 c 程序的栈需求
strings	在目标文件中查找可打印字符串
strip	删除符号表和重定位字位
time	命令计时
tsort	拓扑排序
unset	取消前一个对 scs 文件施用的 get 命令
val	使一个 scs 文件合法化
xref	c 程序的交叉引用
xstr	从 c 源程序中摘录出字符串
yacc	调用一个编译器—编译器

INTRO(CP)

名称

intro——介绍 XENIX 系统开发命令

描述

本节介绍 XENIX 开发系统中各个命令的用法。每个命令名后附加(CP),以示与 XENEX 操作系统和文本处理系统的区别、使用这些字符,可便于说明文件。例如引用 cc(CP)意味着要读者参见本节中关于命令 cc 的说明。注意,这里字母 C 代表“Command”,而字母 P 代表“Programming”。

格式

除非特别说明,本节描述的命令按下述格式接受选择项和其它参数

命令名[选择项][命令参数]

命令名 文件名或一个可执行文件的路径名。

选择项 单个字母代表命令选择项。约定:大多数选择项跟在一个短线后面,有时选择项字母可连在一起(例如-abcd)或各自单独说明(例如-a-b-c-d),指定选择项的方法由各个命令的格式决定,作为另一种指定选择项的方法,参数也可在选择项中给出,例如,很多命令用选择项-f,紧接着的文件名作参数。

参数 路径名或其他参数名不以短划线开头,单独一个短划线表示从标准输入(例如键盘)读入。

参见

getopt(C)、getop(S)

诊断

每一个命令返回两字节的状态标志,其一是由系统提供的,用于说明命令终止执行的原因,另一(正常结束时)由程序(参见 wait(S)和 EXIT(S))提供。前一字节是 0 时表示正常结束,后一字节通常以 0 表示成功执行,以非 0 表示错误(例如参数错、或不可访问的数据),这一字节在不同场合分别称作“退出码”(exitcode)、“退出状态”(exit status)或“返回码”(return code),对它的描述在使用特殊约定时给出。

注释

并非所有命令都遵从上述格式

ADB(CP)

名称

adb——调用一个通用调试器

格式

adb [-w] [-p 提示] [目标文件] [core 文件]

描述

“adb”是一个通用调试程序,它可用于检查文件并提供 XENIX 程序执行的控制环境。