Whitesmiths, Ltd.

IDRIS INTERFACE MANUAL FOR 8086

Edition: 2.3

Date: December 1984

The C language was developed at Bell Laboratories by Dennis Ritchie; Whitesmiths, Ltd. has endeavored to remain as faithful as possible to his language specification. The external specifications of the Idris operating system, and of most of its utilities, are based heavily on those of UNIX, which was also developed at Bell Laboratories by Dennis Ritchie and Ken Thompson. Whitesmiths, Ltd. gratefully acknowledges the parentage of many of the concepts we have commercialized, and we thank Western Electric Co. for waiving patent licensing fees for use of the UNIX protection mechanism.

The successful implementation of Whitesmiths' compilers, operating systems, and utilities, however, is entirely the work of our programming staff and allied consultants.

For the record, UNIX is a trademark of Bell Laboratories; IAS, RSTS/E, VAX, VMS, P/OS, PDP-11, RT-11, RSX-11M, and nearly every other term with an 11 in it all are trademarks of Digital Equipment Corporation; CP/M is a trademark of Digital Research Co.; MC68000 and VERSAdos are trademarks of Motorola Inc.; ISIS and iRMX are trademarks of Intel Corporation; A-Natural, Idris, and ctext are trademarks of Whitesmiths, Ltd. C is not.

TABLE OF CONTENTS

1. Contents of this Manual

1.1. Introduction.. I - 1

2. The Process Interface

2.1. Executable Object Format.................................. II - 2

2.2. Memory Image.. II - 2

2.3. Core Dump Format.. II - 3

2.4. The IDRIS CPU Scheduler................................... II - 5
- 2.4.1. IDRIS System Clock.............................. II - 5
- 2.4.2. Time Quantums................................... II - 5
- 2.4.3. Time Quantum Allocations........................ II - 6
- 2.4.4. Redetermining Process Priority.................. II - 6
- 2.4.5. Bias Priority Using 'nice'...................... II - 7
- 2.4.6. Tuning the Scheduler............................ II - 7
- 2.4.7. Example Uses of 'nice'.......................... II - 8

2.5. IDRIS Swapping Strategy................................... II - 10

2.6. System Calls.. II - 11
- 2.6.1. Conventions..................................... II - 11
- 2.6.2. Error Codes..................................... II - 12
- 2.6.3. Descriptions of Each Call....................... II - 13

- **brk** - set system break to address........................ II - 14
- **chdir** - change working directory......................... II - 15
- **chmod** - change mode of file.............................. II - 16
- **chown** - change owner of file............................. II - 17
- **close** - close file....................................... II - 18
- **creat** - make new file.................................... II - 19
- **csw** - get console switches............................... II - 20
- **dup** - duplicate file descriptor.......................... II - 21
- **exec** - execute file with arguments....................... II - 22
- **exit** - terminate program execution....................... II - 24
- **fork** - create new process................................ II - 25
- **fstat** - get status of open file.......................... II - 26
- **getgid** - get real and effective groupid.................. II - 28
- **getpid** - get processid................................... II - 29
- **getuid** - get real and effective userid................... II - 30
- **gtty** - get tty status................................... II - 31
- **kill** - send signal to process............................ II - 34
- **link** - create link to file............................... II - 35
- **mknod** - make special inode............................... II - 36
- **mount** - mount filesystem................................. II - 37
- **nice** - set priority...................................... II - 38

open - open file.. II - 39
pipe - set up data pipe.................................... II - 40
profil - set profiler parameters........................... II - 41
read - read from file...................................... II - 42
seek - set file read/write pointer......................... II - 43
setgid - set groupid....................................... II - 44
setuid - set userid.. II - 45
signal - capture signals................................... II - 46
sleep - delay for awhile................................... II - 47
stat - get status of named file........................... II - 48
stime - set system time.................................... II - 49
stty - set tty status...................................... II - 50
sync - synchronize disks with memory....................... II - 51
time - get system time..................................... II - 52
times - get process times.................................. II - 53
umount - unmount a filesystem.............................. II - 54
unlink - erase link to file................................ II - 55
wait - wait for child to terminate......................... II - 56
write - write to a file.................................... II - 57

3. Device Driver Interface

3.1. The Driver Environment.................................... III - 1
3.1.1. Hooking Drivers into IDRIS.............................. III - 2

3.2. Conventions.. III - 3
3.2.1. Standard Header File.................................... III - 3
3.2.2. Resident Header File.................................... III - 3
3.2.3. Block Special Device Header............................. III - 4
3.2.4. Character Special Device Header File.................... III - 5

3.3. Resident Driver Support Functions.......................... III - 6

biops - processor level for block devices................. III - 7
brelse - release buffer.................................... III - 8
cmaptab - parity mapping table............................. III - 9
deq - remove buffer from queue............................. III - 10
deqc - dequeue next character to transmit.................. III - 11
deverr - print device error message........................ III - 12
dmajor - obtain major device index......................... III - 13
dminor - obtain minor device index......................... III - 14
enq - add buffer to list................................... III - 15
enqc - add character to queue.............................. III - 16
fetch - get character from user buffer..................... III - 17
flush - clean out character I/O queues..................... III - 18
getaddr - return buffer address............................ III - 19
getblk - get incore buffer................................. III - 20
gsbyte - protect byte from system memory................... III - 21
iodone - notify Idris of I/O completion.................... III - 22
ioerror - print device error on console.................... III - 23
iotick - account for I/O time.............................. III - 24
movbuf - copy buffer....................................... III - 25
nodev - illegal device entry point......................... III - 26

nulldev - innocuous device entry point.................. III - 27
panic - send fatal message and die...................... III - 28
physio - set up character special I/O................... III - 29
putch - put character to console unbuffered............. III - 30
putdnm - print device name on console................... III - 31
putfmt - print formatted messages to console............ III - 32
setch - send character to user buffer................... III - 33
settyp - offer to be controlling terminal............... III - 34
signal - send kill signal............................... III - 35
sleep - wait for event.................................. III - 36
spl - set arbitrary processor level..................... III - 37
spl0 - enable all interrupts............................ III - 38
spl7 - disable all interrupts........................... III - 39
timeout - interrupt and call function after specified time................................ III - 40
ttin - put character on input list...................... III - 41
ttread - transfer characters to user buffer............. III - 42
ttrstart - restart terminal after delays................ III - 43
ttset - complete stty processing........................ III - 44
ttwrite - start transmission from user buffer........... III - 45
ttyps - processor level for terminal devices............ III - 46
uerror - set or test for user error..................... III - 47
vector - redirect 8086 interrupt........................ III - 48
wakeup - post event for all waiters..................... III - 49
wflush - wait for tty I/O to drain...................... III - 50

3.4. Diskdriver Tutorial.................................... III - 51
3.4.1. Establishing Driver Entry Points................. III - 51
3.4.2. Block I/O Open................................... III - 52
3.4.3. Block Device Close............................... III - 53
3.4.4. Device Name...................................... III - 54
3.4.5. Strategy for Block I/O........................... III - 54
3.4.6. Raw Device Support............................... III - 57
3.4.7. Polling.. III - 60

3.5. Terminal Driver Tutorial............................... III - 64
3.5.1. Establishing Terminal Driver Entry Points........ III - 64
3.5.2. Opening Terminal................................. III - 64
3.5.3. Terminal Close Routine........................... III - 66
3.5.4. Read and Write for Terminal...................... III - 66
3.5.5. STTY/GTTY Call................................... III - 67
3.5.6. Terminal Driver Name............................. III - 68
3.5.7. Receiver Interrupts.............................. III - 68
3.5.8. Transmitter Interrupts........................... III - 69

Appendix

a.s - details of CO-IDRIS startup........................ A - 1
Adding a Clock.. A - 2
a.s code - starting up CO-IDRIS......................... A - 3
aca.c - IBM PC-XT serial line driver.................... A - 6
aca.c code - asynchronous communication adapter......... A - 9

bio.h - driver header file used by all drivers........... A - 16
bio.h code - blocked I/O header file..................... A - 17
cio.h - driver header file used only by terminal
drivers.. A - 19
cio.h code - character I/O............................. A - 21
dos13.s - description of IBM PC hard disk I/O........... A - 23
dos13.s code - IBM PC hard disk I/O..................... A - 24
main.c - file of system configuration parameters......... A - 25
main.c code - system configuration parameters............ A - 28
pchd.c - description of IBM PC-XT hard disk driver....... A - 33
pchd.c code - hard disk handler.......................... A - 37
res.h - header file used by all device drivers........... A - 42
res.h code - IDRIS header file.......................... A - 43

1. CONTENTS OF THIS MANUAL

This manual describes the IDRIS operating system in terms of its interfaces to processes running under it and to devices under its control. Our intent is to support installer/implementer efforts at tuning the IDRIS environment to meet specific needs relative to applications programs and embedded system development. We assume that readers are already familiar with CO-IDRIS itself, and with the Whitesmiths, Ltd. documentation accompanying the basic CO-IDRIS package (i.e. the IDRIS Programmers' Manual, IDRIS Users' Manual, and the CO-IDRIS Installation Guide).

Section II concerns the process interface. It describes the following:

1.) format conventions for executable files

2.) program representation in memory

3.) how the IDRIS swapper functions

4.) the IDRIS scheduler and how to tune it

5.) a synopsis of IDRIS error return codes

6.) system call conventions and use

7.) manual page descriptions of each system call

Section III describes the construction of new device drivers. It covers:

1.) the rationale behind IDRIS support of peripheral devices

2.) hooking device drivers into the resident

3.) driver file and header conventions

4.) manual page descriptions of all the routines that the resident provides to support driver functions

5.) a step-by-step tutorial on the construction of those parts of a device driver that are common to any block special device under IDRIS

6.) a step-by-step tutorial on the construction of those parts of a terminal driver that are common to any character special device under IDRIS

The Appendix contains line-by-line explanation of the 8086-dependent code within the resident, and is geared to help the installer reconfigure IDRIS to suit special needs relative to the 8086 architecture. Explanation of the code for a given "system generation" parameter precedes a commented listing of the code itself. (The necessity of flipping back and forth at times will hopefully be offset by the format of code presentation.) Topic areas covered are:

1.) details of CO-IDRIS startup under DOS

2.) the IBM PC-XT serial line driver (asynchronous communications adaptor)

3.) driver header files for blocked I/O

4.) driver header files for character I/O

5.) driver header file used by all drivers

6.) IBM PC BIOS hard disk I/O

7.) IBM PC hard disk driver

8.) IBM PC ROM hard disk driver

9.) system configuration parameters