LDOS (tm)LSI
UTILITY DISKS

By: KIM WATT

from

POWERSOFT

a division of Breeze/QSD, Inc.

UTILITY DISK #2 (c)1982

$29.95

Disk #2 Includes:

PMOD/CMD - A Disk-File-Memory Modification Utility.

PMOD,aaaa (display memory address)
PMOD,filespec,aaaa (file to display,starting rel. sector)
PMOD :b,c,d (drive,cylinder,sector to display)

LOADING INSTRUCTIONS

- MOD I or III -

1) Prepare a SINGLE density LDOS formatted diskette. This disk may be 35, 40, or 80 tracks, but SINGLE density!
2) Boot the MASTER disk you received in this package in drive 0. You will be prompted to insert the destination diskette, and the files will be "moved" to YOUR diskette with no effort on your part. Single drive owners will need to swap disks a few times.
3) LX-80 users should treat the enclosed MASTER disk as a 35 track single density data diskette.

PMOD

The PMOD Utility is a comprehensive Disk/File/Memory Modification Utility. It allows easy user interaction between memory and disk storage systems. All LDOS supported disk devices may be operated on, including Single/Double Density, Single/Double Sided, 5" and 8" floppies and fixed/removable rigid disk systems. It has a full screen display of 256 bytes at a time with both HEX and ASCII formats, and makes use of dual cursors for easy modification. The user can go to the heart of a disk and modify in HEX, ASCII, DECIMAL, BINARY, or OCTAL input. The syntax of the PMOD command is:

```
=====================================================
!  PMOD,aaaa                                        !
!  PMOD,filespec,bbbb                               !
!  PMOD,:c,d,e                                      !
!                                                   !
!       aaaa = memory address to modify             !
!       filespec = any valid LDOS filespec          !
!       bbbb = relative file sector (optional)      !
!       :c = drive to modify (Ø-7, colon mandatory) !
!       d  = cylinder number (defaults Ø)           !
!       e  - sector number (defaults Ø)             !
!                                                   !
!  abbr: NONE                                       !
=====================================================
```

NOTE: It is recommended to SET,*KI,KI/DVR (jkl) before using entering PMOD so the control keys and screenprinter will be active.

There are 3 modes of operation of the program as indicated by the command line:

- Memory Modify -

Enter the memory address where modification is to begin. You may enter the number in Hex, Decimal, Octal, or Binary by PRECEDING the number with H, D, O, or B respectively. Decimal is the default value if none of the above are specified.

After entering the address and pressing <ENTER>, you will view the contents of 1 page of memory (256 bytes) starting at the specified address. Please refer to the following screen dump:

```
    0000 F3AF C315 30C3 0040 C300 40E1 E9C3 1230
HEX0010 C303 40C5 0601 182E C306 40C5 0602 1826
MEM0020 C309 40C5 0604 181E C30C 4011 1540 18E3
    0030 C30F 4011 1D40 18E3 C312 4011 2540 18DB
    0040 C3D9 05C9 0000 C374 06CD 2B00 B7C0 18F9
    0050 11E5 4118 BE11 ED41 18C1 11F5 4118 BC00
    0060 C3FB 0120 FBC9 C339 30C3 5204 111D 4218
    0070 AA00 C3CC 0611 8040 21F7 1801 2700 EDB0
    0080 21E5 4236 3A23 7023 362C 2322 A740 112D
    0090 0106 1C21 5241 36C3 2373 2372 2310 F706
    00A0 1536 C923 2323 10F9 21E8 4370 31F8 42CD
    00B0 BF1B 0000 0021 0501 CDA7 28CD B31B 38F5
    00C0 D7B7 2012 214C 4423 7CB5 281B 7E47 2F77
    00D0 BE70 28F3 1811 CD5A 1EB7 C297 19EB 2B3E
    00E0 BF46 77BE 7020 CE2B 1114 45DF DA7A 1911
    00F0 CEFF 22B1 4019 22A0 40CD 4D1B 2111 01C3
```

The far left column of the screen indicates the current modify BASE (HEX, DECimal, ASCii, OCTal, BINary), and the MEM indicates that you are in the MEMORY MODIFY mode. The next column is the 4 character HEX address where the data is that you are viewing. The center columns of the display is the HEX representation of the 16 bytes starting at the address indicated at the row beginning. There is a space between each quad (2 bytes, 4 HEX ASCII characters) for easier viewing. On the far right is the representative ASCII character of the 16 bytes on that row. All bytes less than 20H (32 decimal) are displayed as a period (.) to indicate a non-printing character. You are now in the MEMORY PAGING MODE. Several keys are active at this point:

BREAK - terminate the program
SHIFT CLEAR - prompt for a new address
M - enter MODIFY mode
A - set ASCII modify
B - set BINARY modify
D - set DECIMAL modify
H - set HEXIDECIMAL modify
O - set OCTAL modify
X - prompt for new source data
RIGHT ARROW - increment displayed address by 1 byte
LEFT ARROW - decrement displayed address by 1 byte
UP ARROW - increment displayed address by 1 page (256 bytes)
DOWN ARROW - decrement displayed address by 1 page

- Disk Modify -

Enter a Drive, Cylinder, and Sector where modification is to begin. You MUST precede the drive number with a colon (:) to distinguish between a memory address. The Cylinder and Sector numbers default to 0 if not issued. The Cylinder may be answered with the AT symbol (@) to page directly to the directory track.

After answering the prompt and pressing <ENTER> the specified sector will be read from disk and displayed to the video. Please refer to the

```
     00 00FE 14F3 3A02 4357 1E04 CD74 4320 713A .▗.▝:.CW..▙tC q:
HEX  10 0051 E610 286E D92A 1651 557C 0707 07E6 .Q▚.(n▚*.QU¦...▞
DRV  20 0767 0784 075F 21FF 51D9 CD5D 433D 200C .g.▚._!▝Q▙]C= ▗
 0   30 CD4E 43CD 5D43 7723 10F9 18EE 3D28 0BCD ▙NC▙]Cw#.▚▗▗=(.▙
CYL  40 5D43 47CD 5D43 10FB 18E0 CD4E 43E9 CD5D ]CG▙]C.▚▗ ▙NC▜▙]
 00  50 4347 CD5D 436F 05CD 5D43 6705 C9D9 2C20 CG▙]Co.▙]Cg.▜▚,
SID  60 0FE5 CD74 4320 19E1 1C7B D612 2002 5F14 .▌▙tC .▚.{▚. ._.
 0   70 7ED9 C9C7 0605 C5CD 8E43 C1E6 1CC8 10F6 ~▜▞..▌▙C▝▚.▗.▞
SEC  80 21E6 43DD 21F4 43CD 1B02 CD40 00C7 01F4 !▗C▌!▗C▙..▙@.▞.▗
 00  90 B1ED 410D 3E18 ED51 CDDF 43DB F0CB 4720 ▝▙A.>.▙Q▙▙C▜▙▝▙G
STD  A0 FA7B D3F2 3E81 D3F4 21D3 4322 4A40 3EC3 ▙{▞▞>▝▙▙▗!▗C"J@>▞
 5"  B0 3249 4021 0051 D511 02C1 3E80 CDDF 433E 2I@!.Q▙..▝> ▙▞C>
FLP  C0 C0D3 E4DB F0A3 28FB EDA2 7AD3 F4ED A220 ▞▗▙▗▙▗(▞▙▞z▞▙▗▝▗
 DD  D0 FA18 FEE1 D1AF D3E4 3E81 D3F4 DBF0 C9D3 ▙▗▝▞▙▞▞▗▞>▝▗▙▞▙▞▞
 SS  E0 F006 0C10 FEC9 17E8 4469 736B 2065 7272 ▞...▞▗▗ Disk err
     F0 6F72 1F03 17E8 4E6F 2073 7973 7465 6D03 or... No system.
```

The screen format is similar to Memory Modify except for the far left column of data, which indicates the source of the information. You are given the Drive, Cylinder, and Sector being viewed. You are supplied with the size of the drive (5 or 8"), and if it is a Floppy or Rigid drive. If a floppy, you are given the density (single or double), and the number of sides (single or double). If a Rigid drive, you are told if it is Fixed or Removable. You are now in the SECTOR PAGING MODE. The following keys are active:

BREAK - terminate program
SHIFT CLEAR - prompt for new Drive, Cylinder, and Sector
M - enter MODIFY mode
A - set ASCII modify
B - set BINARY modify
D - set DECIMAL modify
H - set HEX modify
O - set OCTAL modify
X - prompt for new source data
0-9 - page to corresponding sector
@ - page to current sector on directory track
R - restore drive to cylinder 0, sector 0
RIGHT ARROW - increment sector
LEFT ARROW - decrement sector
UP ARROW - increment track
DOWN ARROW - decrement track
SHIFT RIGHT ARROW - page to highest sector on current track
SHIFT DOWN ARROW - page to lowest sector on current track
SHIFT UP ARROW - page to highest cylinder on disk, current sector

- File Modify -

Enter a filespec that you wish to view. You may optionally follow the filename with the relative sector that you wish to start with. After locating the file, the specified relative sector (default zero) will be read from the disk and displayed to the video. Please refer to the following screen dump:

```
     P00 0506 504D 4F44 2020 0102 0052 E521 A462 ..PMOD  ...R.!.b
HEX  M10 CD67 44E1 1807 2100 6436 0D18 EFCD 6F53 .gD"..!.d6..  oS
DRV  O20 DA48 53FE 3F20 08E5 CD7C 60E1 2318 EEFE  HS.? .  !@"#.
 0   D30 3A28 19FE 21CA EA52 E5CD 8253 E1DA EB52 :(.  ! R  "S  R
     /40 ED43 EE55 3E01 324C 56C3 2155 237E D630 C.U>.2LV=!U#~.0
     C50 DA39 53FE 08D2 3953 3268 524F E53A 0400  9S.  "9S2hRO :..
     M60 FE30 2805 CDB8 4418 03CD 9042 3E08 E1C2 0(.  D..  B>.  "
     D70 3B53 E53E 004F CD8F 47FD 220C 5EE1 23AF ;S >.O  G  ".^" #
      80 32CF 5D32 CE5D FD7E 07E6 1F3C FDCB 046E 2 ]2 ]  ~.  .<  .n
      90 2802 CB27 FDCB 035E 2811 4FFD 7E07 0F0F (.  '  .^(.O  ~...
      A0 0F0F 0FE6 073C 47AF 8110 FD32 C354 CD6F ....  .<G  .  2 T o
      B0 5338 23FE 4020 06FD 7E09 2318 07CD 8253 S8#  @ .  ~.#..  "S
RSEC  C0 DA42 5379 32CF 5DCD 6F53 380A CD82 53DA  BSy2 ]  oS8.  "S
0000  D0 4253 7932 CE5D 3E02 324C 56C3 9C5D 21B9 BSy2 ]>.2LV=  ]!
EOFS  E0 61CD 6744 061F 2100 64CD 4000 DA65 5378 a gD..!.d @.  eSx
0018  F0 B7CA 0A52 1832 2311 5F64 CD1C 44C2 3B53  .R.2#._d .D ;S
```

The screen layout is similar to the Memory Modify, except for the far left column. You are given the drive number that you are working on. The filename that you are viewing is displayed vertically starting just to the right of the cursor. Near the bottom you are given the RELATIVE SECTOR you are viewing, and the END OF FILE SECTOR. Normally, you will only be able to view up to 1 less that the EOFS. You are now in the FILE PAGING MODE. Several keys are active:

BREAK - terminate program
SHIFT CLEAR - prompt for a new relative sector
M - enter Modify mode
A - set ASCII modify
B - set BINARY modify
D - set DECIMAL modify
H - set HEX modify
O - set OCTAL modify
X - prompt for new source data
RIGHT ARROW - increment to the next relative sector
UP ARROW - same as right arrow
LEFT ARROW - decrement to next relative sector
DOWN ARROW - same as left arrow

NOTE: When using the FILE MODIFY option on DIR/SYS, the sectors will NOT be written as Read-Protected, and will require a "Repair (alien)" after updating back to disk. Use the DISK MODIFY option to correctly handle the directory track.

- Modify Mode -

The MODIFY MODE is common to all of the above 3 modes of operation, and is entered by pressing M when the desired data is being viewed in one of the above PAGING modes. The cursor will move into the HEX and ASCII portions of the display (dual cursors), and will flash very rapidly. Refer to the following screen dump:

```
67]  ØØ ØØFE 14F3 3AØ2 4357 1EØ4 CD74 432Ø 713A .▐.▝=:.CW..▙tC q:
HEX  1Ø ØØ51 E61Ø 286E D92A 1651 557C Ø7Ø7 Ø7E6 .Q▚.(n▚*.QU|...▚
DRV  2Ø Ø767 Ø784 Ø75F 21FF 51D9 CD5D 433D 2ØØC .g..._!▚Q▚]C= .
 Ø   3Ø CD4E 43CD 5D43 7723 1ØF9 18EE 3D28 ØBCD ▙NC▙]Cw#.▚..▚=(.▙
CYL  4Ø 5D43 47CD 5D43 1ØFB 18EØ CD4E 43E9 CD5D ]CG▙]C.▚.▚▚NC▚▚]
 ØØ  5Ø 4347 CD5D 436F Ø5CD 5D43 67Ø5 C9D9 2C2Ø CG▙]Co.▙]Cg.▚▚,
SID  6Ø ØFE5 CD74 432Ø 19▚▚ 1C7B D612 2ØØ2 5F14 .▐▙tC .▚.{▚. ._.
 Ø   7Ø 7ED9 C9C7 Ø6Ø5 C5CD 8E43 C1E6 1CC8 1ØF6 ~▚▚..▐▙C▚.▚..▚
SEC  8Ø 21E6 43DD 21F4 43CD 1BØ2 CD4Ø ØØC7 Ø1F4 !▚C▚!▚C▚..▙Ø.▚.▙
 ØØ  9Ø B1ED 41ØD 3E18 ED51 CDDF 43DB FØCB 4720 ▚▙A.>.▙Q▙▚C▚▚▚G
STD  AØ FA7B D3F2 3E81 D3F4 21D3 4322 4AAØ 3EC3 ▙{▚>▚▚▚▚!▚C"JØ>▚
 5"  BØ 3249 4Ø21 ØØ51 D511 Ø2C1 3EØØ CDDF 433E 2IØ!.Q▙..▚> ▙▚C>
FLP  CØ CØD3 E4DB FØA3 28FB EDA2 7AD3 F4ED A22Ø ▚▚▚▚▚(▚▚z▚▚▚▚▚
 DD  DØ FA18 FEE1 D1AF D3E4 3E81 D3F4 DBFØ C9D3 ▚▚▚▚▚▚▚▚>▚▚▚▚▚▚▚
 SS  EØ FØØ6 ØC1Ø FEC9 17E8 4469 736B 2065 7272 ▚...▚▚▚▚ Disk err
     FØ 6F72 1FØ3 17E8 4E6F 2Ø73 7973 7465 6DØ3 or... No system.
```

At the top left of the display, you are given the RELATIVE BYTE within the displayed data where the cursor lies. The following keys are active at this point:

BREAK - abort the modify mode, re-read the source data as it was before any changes were made, and return back to the paging mode you came from. NOTE: In the MEMORY MODE, the changes are immediate, and cannot be re-read. In this case, the BREAK key is identical to pressing the ENTER key.
ENTER - terminate the modify mode, and write the modified data back to the source with all changes, and return to the paging mode you came from. NOTE: Changes are immediate in the MEMORY MODE, and control is merely passed back to the paging mode.
SHIFT CLEAR - homes the cursor to the first relative byte being displayed.
CLEAR ENTER - allows you to change the current BASE you are modifying in without leaving the modify mode. You will have a new flashing prompt in the left column of the display. Enter A, B, D, H, or O to set ASCII, BINARY, DECIMAL, HEX, or OCTAL bases respectively. Any other keys will be ignored. You will then return back to the modify mode at the same location you came from.
ARROW KEYS - move the cursor 1 byte in the corresponding direction without affecting any of the bytes.
SHIFT ARROW KEYS - move the cursor to extreme row or column ends in the associated direction without affecting any of the bytes.

NOTE: In the DISK MODIFY and FILE MODIFY you will not be able to move the cursor beyond the edges of the displayed data. In the MEMORY MODIFY, if you attempt to move the cursor beyond the edges, additional data is brought to the screen to supply the desired information.

If you are in the ASCII MODIFY MODE, and the key you enter does not apply to any of the above conditions, then that character is entered at the current cursor location, and the cursor is advanced by one byte.

If you are NOT in ASCII MODIFY, several additional features are available:

G - followed by a number moves the cursor immediately to the RELATIVE BYTE in the page that you are viewing.
L - followed by a number moves the cursor to the NEXT OCCURENCE of the specified byte. The cursor will move to the last byte on the screen if the requested byte is not located.
P - followed by a number DUPLICATES the current byte located under the cursor to the following indicated number of bytes.
< - will shift all bytes from the cursor to the page end by one byte, and place a zero at the last location on the screen. This allows you to DELETE text at the cursor.
> - will shift all bytes from the cursor to the page end by one byte, and place a zero at the current cursor location. This allows you to INSERT text at the cursor.

NOTE: The above five commands are disabled in the ASCII MODIFY MODE, and will merely insert the character into the text.

If you are NOT in ASCII MODIFY, and NONE of the above conditions are met, then the input is interpreted as part of NUMERICAL input. If you are in HEX mode, for example, you must enter TWO HEX DIGITS (Ø-9,A-F) to change a single byte. Decimal mode expects 3 digits (Ø-9), Octal mode expects 3 digits (Ø-7), and Binary mode expects 8 digits (Ø-1). You may pre-terminate the numerical input by pressing ENTER. Thus, in HEX mode, 3 ENTER is the same as entering Ø 3, in Binary mode, 1 1 1 ENTER is the same as entering Ø Ø Ø Ø Ø 1 1 1. When you are in the middle of entering a number, the cursor will change to indicate that more input is required to complete the current operation. If you enter incorrect digits BEFORE completing the number, entering an INVALID character will terminate the entry. Thus, in HEX modify, you type a 3, but meant to enter a 4, just hit an invalid key (X for example) to terminate the operation and leave the byte unchanged. In the above special commands, G, L, and P, the numerical input following the command will be in the current BASE. If you are in HEX modify, then G1Ø is a valid command to move the cursor to relative byte 1ØH, but in DECIMAL modify, you must enter GØ16 to achieve the same results.

Any key that does not meet any of the above conditions will be ignored.

When first entering this program, the DATA SOURCE information may be entered directly from the DOS READY command line, or will be prompted for if not supplied with the command. If you wish to examine a file that could be interpreted as an address, you may PRECEDE the filename with an exclamation point (!) to FORCE PMOD to interpret it as a filespec.

PMOD,HFØØØ	will display memory address FØØØH.
PMOD,!HFØØØ	will display the file HFØØØ

LDOS (tm)LSI UTILITY DISKS

By: KIM WATT

from

POWERSOFT

a division of Breeze/QSD, Inc.

UTILITY DISK #2 (c)1982

$29.95

Disk #2 Includes:

PMOD/CMD - A Disk-File-Memory Modification Utility.

PMOD,aaaa (display memory address)
PMOD,filespec,aaaa (file to display,starting rel. sector)
PMOD :b,c,d (drive,cylinder,sector to display)

UTILITY DISK #3 (c)1982

$29.95

Disk #3 Includes:

PCHECK/CMD - LDOS Directory Check Utility. Checks a drive or a file, (GAT, HIT, BOOT, etc.) for integrity.
PFIX/CMD - FIXES problem discovered by PCHECK/CMD. You may repair GAT or HIT table, repair BOOT sector, or repair FILE records. This program might fix that crashed disk that has you going nuts!

UTILITY DISK #4 (c)1982

$29.95

Disk #4 Includes:

PFIND/CMD - FIND a string, replace a string, find a byte, find a WORD, replace a byte or word! FIND ANYTHING on the disk FAST! Replace it FAST!
PCOMPARE/CMD - Compare a FILE or a sector to any other file or sector. Detects the minutest differences! Send the outputs from above to video or printer.

UTILITY DISK #5 (c)1982

$29.95

Disk #5 Includes:

PREFORM/CMD - Disk Re-Formatter Utility. Fixes CRC errors and "stengthens" the format on older disks. A FORMAT without ERASE utility(5" floppy only)
PVU/CMD - a Disk Verification Utility. Detects bad sectors. Fix them with PREFORM/CMD.
PERASE/CMD - Disk Bulk Eraser (5" floppy only)

UTILITY DISK #6 (c)1982

$29.95

Disk #6 Includes:

PCLEAR/CMD- Erases unassigned sectors and Directory slots. Removes ALL traces of "killed" files!
PSS/CMD - Sector Status Utility. Finds WHAT file is assigned to WHAT sector.
PMAP/CMD - Disk-File Mapping Utility. Map a FILE or a disk! Shows exactly WHERE all files are with extents.

UTILITY DISK #7 (c)1982

$29.95

Disk #7 Includes:

PMOVE/CMD - A SUPER FAST Multiple Copy Utility!
PDIRT/CMD - Read a Mod III TRSDOS directory from LDOS!
PASSGO/CMD - Removes passwords from A file or a WHOLE disk!
PUN/CMD - UN-REPAIR a disk. (Mod I only)
PEX/CMD - A disk exerciser for head cleaning kits.

UTILITY DISK #8 (c)1982

$29.95

Disk #8 Includes:
PMX/FLT-MX80 graphics
PHELP/CMD -Instant HELP! Very complete. Saves time!
PBOOT/FIX - Customize YOUR LDOS Boot graphics and message!
PFILT/FLT - A User Definable Filter. Very flexible!
DVORAK/FLT - DVORAK/QWERTY Keyboard Filter.
DVORAK/JCL - DVORAK Keyboard Table (used with PFILT/FLT)
CODE/JCL - Keyboard EN-Coding TABLE (used with PFILT/FLT)
DECODE/JCL - Keyboard DE-Coding Table (used with PFILT/FLT)