PROSEL (PROgram SELector)

By Glen E. Bredon

TABLE OF CONTENTS

Main documentation:

Miscellaneous:

Backup, restore and RAM drive doc:

Disk utilities:

Documentation for ProSel (PROgram SELector) version 4.0

by Glen E. Bredon

ProSel is a sophisticated "program selector" which operates under ProDOS. It lets you pass easily between applications without the repetitive typing of prefixes and pathnames usually associated with quitting from an application program running under ProDOS. Programs are selected from a list of titles supplied by you. You do not have to remember where you put these applications on your hard disk, ProSel remembers for you. ProSel can also be used to boot BASIC programs with names other than "STARTUP". It can allow you to select from among all the SYS type files on a given directory. It supports the Apple mouse, but does not require it, and works very well without it.

There are three versions of ProSel on this disk. The installation programs will ask which you want. They differ in that one has an 80 column display, another a 40 column display, and the third designed for use on a][+ with a Videx Videoterm card. The 40 column version has more space for application specification, but allows fewer of them. Also the 40 column version is a little faster. Choose the one that suits your fancy. You can change them by running the INSTALL.REVISN program later. I recommend the 80 column version which has more features.

INSTALLING PROSEL

ProSel can be installed on hard disk, 800k disks and /RAM volumes with little effort. If you use 800k disks, you may want to install ProSel on all disks that have application files. This can be done by simply copying the files PROSEL.SYSTEM and PROSEL from one disk to another after the first installation. The INSTALL.PROSEL program installs a number of other utilities and you may not want these on some disks. You can decide about this later. For now, just follow these directions:

It will be assumed that the disk to receive ProSel has been ProDOS formatted. (Eg., that hard disk installation has been done.)

1. Turn on the computer. Boot the supplied disk called /FLOPPY.

2. Select item #1 from the menu "Install new ProSel system". You will be asked for the name of the receiving volume. You can just press RETURN to accept the default name shown. Please note that installation will not make any irreversible changes.

3. If no errors occur you will be sent directly to the ProSel menu.

4. If the install program encounters an error you will be told what you must do to correct it.

This will establish the files PROSEL.SYSTEM, PROSEL, and BASIC.SYSTEM on your root directory. If you do not want BASIC.SYSTEM there you can delete it later, but I suggest you leave it there until you are familiar with the operation of ProSel. It will also create the subdirectory UTIL and install most of the utilities in it.

ProSel comes with several application specifications already set up. These items are supplied to demonstrate the system. You will probably want to change them all later.

GETTING STARTED (Simplified instructions)

One of the first things you will want to do is to enter new items into the "desktop". There are several ways to do this, but the one recommended to beginners is the one that uses the "external editor". That editor works only with the 80 column version of ProSel, so this discussion assumes that is the one you installed.

1. After installation and when you are "in" ProSel, look on the screen for the item "ProSel editor". Use the arrow keys or the mouse to highlight that item. Then press RETURN or the mouse button.

2. The disk should then read the PROSEL.ED file and run it. You should see a screen that is similar to the ProSel screen but which has the text "PROSEL EDITOR" at the bottom.

3. Use the arrow keys (the mouse is not supported in the editor) to highlight an EMPTY space on the screen, and then press RETURN.

4. You should now see a screen with some instructions and a choice of "automatic" or "manual". Choose automatic.

5. This will give a "Search what directory" message and a default directory name which is just your volume name (the volume from which the editor was run). Just press RETURN.

6. The disk will operate for a short while. (The editor is reading the entire directory including subdirectories.)

7. In a short time, a list of file names will be displayed on the screen. These are all the directly executable files found by the editor. (If the screen looks full then some executable files will be missing and you would have to use Manual mode to deal with them, or specify a subdirectory in step 5.)

8. Use the arrow keys to highlight a program you wish to add to the desktop. For example, Appleworks would probably appear on the screen as APLWORKS.SYSTEM. Then just press RETURN.

9. You should now see the ProSel-like screen again, but with the application title you just chose. The name may look a little different. For example, APLWORKS.SYSTEM would come out as "Aplworks System".

10. If you do not like the name the automatic mode created for the application, then press RETURN with that title highlighted. This will present another screen which is the "manual" mode screen of the editor. The cursor will be on the first of the four items displayed in boxes, the "Screen title". Type over this with the name you want it to have. Use control-Y to clear the line from the cursor on. DO NOT modify the other 3 items. When you are done, press the ESC key.

11. You should now be back in the main editor screen (the one that looks like the ProSel screen). Press the ESC key.

12. You will be given some options. The last option "P" is to save the modified ProSel file, and that is the option you want, so press the P key.

13. The disk will operate and the modified ProSel file will be written to it, and you will be sent back to ProSel (not the editor). You should see the item you added on the screen.

14. Select that added item by highlighting it and pressing RETURN or the mouse button. This should run that application.

15. Quit from that application. (Select Quit or Exit or Shutdown, whatever the application calls that.) This should send you back, very fast, to ProSel again.

16. Press the "1" key. This will bring up a list of the executable files in the main directory of the "drive 1" volume on the logged slot, and also the subdirectories (shown in lower case). Select one of these. If it was a subdirectory, it will now show the executable files and subdirectories there. This shows how you can access files not on the main selection screen. Press ESC if you don't want to run any of them. If you have a disk in slot 6 drive 1, try pressing the 61 key sequence and note that this allows access of files on that disk.

17. You are now up and running. However, ProSel has many parts, most of which are disk utilities. Examine these, and read about them here at your leisure. I would suggest that you read about, and operate, the main utility called Cat Doctor. That is a file maintenance program which is very powerful. I think you will find that program indispensable. It is quite friendly, but you should read the documentation on how to make "selections" and a few other things that are not indicated on its screen.

18. This about covers the ordinary everyday usage of ProSel. Unless you have something specific you are anxious to do, like using the disk repair program Mr.Fixit, you can leave the rest aside for now and examine and experiment with the rest of the utilities at your leisure.

APPLICATION SPECIFICATIONS AND PATHNAME CONVENTIONS

If you have accessed the manual mode in the external editor, you have noted that a ProSel "application specification" consists of four items:

A screen title:

> This can be anything, and may have very little connection to the name of the application. It may also contain some "command characters" which tell ProSel to do special things when showing the title on the screen.

A prefix:

> This indicates the directory (folder) which will be open when the program is run. Usually you would set this to the directory containing the application program. For example, for Appleworks on a disk called /HARD1/, it might be /HARD1/APPLEWORKS or ?APPLEWORKS which means the same thing to ProSel.

An application name:

> This is the pathname of the program. It can be either a FULL pathname (which means it begins with a /) and the volume name (eg., /HARD1/APPLEWORKS/APLWORKS.SYSTEM), or a PARTIAL pathname (meaning it does NOT start with a / or ?) and will be ADDED to the prefix to form a FULL pathname. For example, it could be APLWORKS if the prefix is given as /HARD1/APPLEWORKS (or as ?APPLEWORKS).

A startup:

> A "startup" is an instruction to the program. It might be a path to some file you want opened, or, in the case of the application BASIC.SYSTEM, it would be the name of the BASIC program (BAS or BIN or TXT filetype) you want run immediately. Except for BASIC.SYSTEM and programs whose documentation indicate they use a startup, this item can be ignored and left blank.

Besides the usual ProDOS filename syntax, ProSel supports some abbreviations that not only save space in its list, but ease use of ProSel on multiple volumes. The most important abbreviation, and one I recommend using almost always, is the use of the question mark character. If a question mark appears in any of the three items PREFIX, APPLICATION NAME, or STARTUP, then it is interpreted as the volume name INCLUDING the slash on BOTH ends. Thus, on a volume called /HARD1,

```
? = /HARD1/     and

?UTIL = /HARD1/UTIL
```

A second space saver is the use of the right bracket (the Applesoft prompt character) as shorthand for BASIC.SYSTEM. This abbreviation does NOT include a slash on either side.

THE INTERNAL EDITOR

There are actually two editors available for ProSel: First, the external editor, described briefly in the last section, and in more detail later. That is a separate program you RUN from ProSel. Second, there is an internal editor which is not as powerful but is sometimes convenient to use.

When you boot you will be sent immediately to application select mode. You can press the ESCAPE key to go to the EDIT menu to edit items in the list, add items to it, delete items, and save the modified PROSEL file.

The internal editor provides the following options. The EDIT, ADD, and DELETE options are recursive and you have to press the ESC key to return to the menu.

QUIT: This exits the editor and sends you to application select mode.

EDIT: This prints the list of the application programs in the selector. You are expected to use the arrow keys (or mouse) to highlight one to edit and then press RETURN (or the mouse button). (Pressing ESC will send you back to the menu.) After you select, the editor will print the existing title of the application (what ordinarily appears on the screen). You should modify this according to your wishes. (Such titles must not exceed 23 characters. They may contain spaces.) Then the prefix belonging to the application appears and you must modify this. Finally, the same is done with the application pathname and the optional startup. If you don't wish to change one of these, trace over it with the right arrow key.

DELETE: This prints the selection list. Highlight the one to be deleted and press RTN (or the mouse button). Use ESCAPE if you entered delete mode by mistake.

ADD: This will be ignored if there are already 54 applications (48 for the 40-column version) listed. The ADD command will request a title, then a prefix, then the pathname of the application, and finally an optional STARTUP program. Two examples are:

```
        Screen title:   Appleworks
        Prefix:         ?APPLEWORKS
        Pathname:       APLWORKS.SYSTEM
        Startup:        (empty)

        Screen title:   MYSTART hello
        Prefix:         ?PROGRAMS
        Pathname:       BASIC.SYSTEM
        Startup:        MYSTART
```

The latter example will cause BASIC.SYSTEM to execute the MYSTART program directly instead of running the usual STARTUP program. If the Startup line had been left blank then the program named STARTUP would be executed as usual.

If the "pathname" is just the single character "/" then selecting this application will result in the directory corresponding to the prefix being read. Then the SYS files in that directory are displayed and you can use the mouse (or arrows) to select one of those to be executed. Pressing ESC at this point will read the original list back in.

If the "startup" is just the single character "]" and the pathname is a valid path pointing to the BASIC.SYSTEM file, then selecting this application will result in the directory corresponding to the prefix being read. Then the BAS files and BIN files will be displayed and you can select one of these. In this case, the file selected will be a "startup" program for BASIC.SYSTEM. The BAS files will be shown in upper case and the BIN files in lower case. (Remember that not all BIN files are executable programs, and ProSel will have no knowledge of this if you select such a program.) Note that the prefix does not have to correspond to a directory containing the BASIC.SYSTEM file, as long as the pathname is a full pathname telling ProSel where to find the BASIC.SYSTEM file.

The prefix must be a valid full pathname of a directory file and the application file must be of SYS or S16 type or it will be ignored. The editor, however, will have no knowledge of this. The startup file may be of any type supported by the application. For example, BASIC.SYSTEM will support any file for which the "-" syntax is acceptable. MERLIN-PRO supports most of its utilities such as UTIL/ED.16, UTIL/XREF, and SOURCEROR/OBJ.

There is only so much space (over 2300 bytes in PROSEL.80, 2600 in PROSEL.40) for the file list. If adding an entry, or editing to make one longer, causes overflow of this space then the addition will be deleted automatically. With modest lengths of names this should not happen. If it should, you can edit some of the titles to make them shorter. The internal editor automatically places an added or edited item in alphabetical order.

Control characters are allowed in screen titles (supported by the external editor) except in the 40-column version. This can be used to make "headers" with inverse or mousetext characters. You must avoid control characters that move the cursor, except backspace (control-H). If a title starts with control-@ then it will be shown in inverse.

Exit the add mode with the ESC key. Some hardware intercepts the ESC character, so I provided that TAB (control-I) also exits from ADD mode in the 80 column version.

SAVE: This saves the PROSEL file, as you have changed it, under the title /XXX/PROSEL where XXX stands for the name of the volume in which ProSel finds itself. CAUTION: Both PROSEL.SYSTEM and PROSEL must be in the main directory. Other files can be anywhere. You should not lock the file PROSEL or else the save will not work. Note that the SAVE command never creates the file, it only overwrites an existing one. It cannot be used to transfer that file to another disk.

USING PROSEL

ProSel comes up when you boot and when you "quit" from an application. For example, the BYE command in BASIC.SYSTEM does this.

When you quit, the PROSEL file is read in from disk and the list of application programs is displayed on the screen. You should use the mouse (or the arrow keys if you don't have a mouse) to highlight the application you want executed and press the mouse button or RETURN.

If you get an "Invalid pathname" error when trying to go to an application, it probably means that the prefix or pathname specification has incorrect syntax. Go to the editor and check it.

When you quit from an application program and the PROSEL file list comes up on the screen, you can elect to go to the ProSel editor to modify the list. To do this, type the ESC key. When you quit from the editor you will go directly to the modified list of application programs and can select one to run. If you have not used the editor's save command then the changes made will be for this time only. If you find yourself in the editor by accident, just select the quit command and you will be sent back to the application selection mode.

It is possible to have BASIC.SYSTEM on one directory and have it boot a startup program in a second directory. The following application specification is an example of this:

```
        Screen title:   MYPROGRAM
        Prefix:         ?
        Pathname:       BASIC.SYSTEM
        Startup:        /VOL2/MYPROGRAM
```

Then BASIC.SYSTEM will be loaded from MYVOL and will run MYPROGRAM in the volume VOL2. In this way you need not have a copy of BASIC.SYSTEM in both volumes.

ADDITIONAL CAPABILITIES

If, in "Select" mode, you press a key "1" or "2" then the volume directory from drive 1 or 2, respectively, of the currently logged slot will be read and the list of SYS files from that directory displayed. You can select one of these to run or press ESC to go back to the regular application list. If you press a key "3" to "7" followed by "1" or "2" then that slot and drive will be selected. Thus, for example, if you press "6" and then "1" the disk in slot 6, drive 1 will be read and its list of SYS files displayed for selection. For those having drives in slots 1 or 2, the keys "8" and "9" are translated to slots 1 and 2. Besides the SYS files, the subdirectories are listed in lower case, and you can select one of those to read that subdirectory.

An alphabetic key pressed in select mode will move the cursor to the next item beginning with that character. If there is none the cursor will return to the first item on the screen.

There is a convention for users of the Ultraterm on the enhanced //e: If an = sign appears anywhere in the prefix or application path, etc., this is taken as a signal for ProSel to turn on the Apple ROMS (so the Apple 80 column card will be active when the application turns on 80 column mode). Holding down the open apple key when selecting does the same. Return to ProSel always reselects the Ultraterm if it is present. If the = sign is followed by a number 1-8, however, then the corresponding mode of the Ultraterm will be switched in prior to executing the application. In this case the application will be entered in 80 column mode. (In all other cases 40 column mode is selected prior to running an application.) Note, however, that some applications (eg. BASIC.SYSTEM) may reset 40 column mode; nevertheless the selected Ultraterm mode will come up when the 80 column card is selected (PR#3 etc.). Although the = sign can be anywhere in the application specification, I suggest that it be put at the end of the prefix.

If the ProSel list has been on the screen for about 5 minutes with no activity then it will blank the screen. Any keypress will bring it back. The time delay can be adjusted by running the INSTALL.REVISN program. At the screen blank, ProSel also parks the heads on the hard drive, if you have one.

The program PRINT.NAMES is a utility that allows you to print out the application specifications in the PROSEL file. This list can be sent to a printer if you want. This may help if you are having trouble getting some application to load properly.

REMARKS

You may find some programs to be incompatible with ProSel. The only reason for this would be that these programs fail to observe ProDOS protocol. You would be well advised to stand clear from such programs. ProSel obeys all established protocol, attempts to circumvent potential problems, and SHOULD work flawlessly.

With respect to the 40 column version of ProSel, you might ask how 48 names are going to fit on the screen in two columns of 24 rows with the text at the bottom. The answer is, it won't. You will find that if your list grows close to this amount then the text at the bottom will start disappearing bit by bit. If this disturbs your sense of propriety, use the 80 column version which does not use or need such machinations.

Some applications will defeat the Ultraterm in order to use the Apple 80 column screen. Most of these do not reestablish the Ultraterm on exit. ProSel will do this for you.

USING PROSEL ON MULTIPLE VOLUMES

This is addressed mainly to those who use ProSel on 800K drives. It may be desirable for you to have different versions of ProSel on several different disks. ProSel is small enough so that it does not use up much valuable space to do this, making this option quite viable. All you have to remember to do when you change disks is to type "51" (assuming the disk is in slot 5, drive 1) with the new disk in the drive and select PROSEL.SYSTEM from the list of files. This is not necessary unless you need the application list on the replacement disk right away. ProSel will renew itself automatically from the new disk. Instead of the "51" you could have an item in your lists called Reboot ProSel (for example) that would have the single character "?" as prefix and "PROSEL.SYSTEM" as pathname. For this provision to work correctly you must use the "?" syntax throughout your application list for the volume name.

On most 800K disks you would want only a minimal configuration of ProSel rather than a full installation. The minimal configuration consists of the files PROSEL.SYSTEM and PROSEL in the main directory. Nothing else is required, but it is recommended that the external editor (file PROSEL.ED) also be put on the disk. Note that there is no PROSEL file on the 5.25" ProSel disks. Instead, one of the files PROSEL.80, PROSEL.40 or PROSEL.VT should be copied to the 3.5" disk and RENAMED to PROSEL. (You can use Cat Doctor to do this.)

USING PROSEL ON A /RAM VOLUME

ProSel can be installed on a large ram card. Just specify /RAM as the receiving volume. Also provided is a /RAM driver for the Multiram and Ramworks cards. This /RAM driver is compatible with Merlin-pro version 2.3 and over. You can use any RAM driver you want to with ProSel, it is not limited to the one on the disk.

Ram cards, such as the Apple memory card and the RamFactor card, that go in a regular slot (1-7) instead of in the aux slot, do not need any driver, and you should ignore references to a ram driver here if you have such a card.

When using ProSel on a RAM disk you will need some efficient means of copying your files between RAM disk and floppy. Although you can use CAT.DOCTOR or another file copying program, that would be slow and inconvenient. Much better would be to use BACKUP and RESTORE to back up the RAM disk to floppy (as soon as you have arranged the RAM disk the way you want it). It is important to note that BACKUP and RESTORE may not work with some RAM drivers, but they do with the one supplied on the ProSel disk.

After you have established ProSel and the other files that you want on the RAM disk and have used BACKUP to make backup disks of RAM, you can automate the restoration process by running the program RAM.RESTORE. (That program uses the BOOT.PROSEL.R program as well as the RESTORE program. Don't use BOOT.PROSEL.R for anything else.)

Note that RAM.RESTORE relates only to backup DISKS. BACKUP also has the ability of backing up to a FILE. For instructions on using that facility, read the section BACKING UP AND RESTORING TO AND FROM A FILE.

Here are detailed instructions for setting up a boot disk to restore the /RAM drive. If you have a hard disk or 800k disk then you should consider using, instead, the facility for backup/restore to/from a FILE. For floppy backups:

(1) First format a floppy disk, calling it, eg., RESTORE.RAM.

(2) Copy the files PRODOS, BASIC.SYSTEM, RAM.RESTORE, BOOT.PROSEL.R, and RESTORE from /FLOPPY to the new disk.

(3) Copy the ram driver that you use to the new disk. Make sure it is a BIN file. (If you use ProSel's RAM driver, there are both BIN and SYS versions.)

(4) Go to BASIC and set the prefix to the new disk. (I.e., Type PREFIX /RESTORE.RAM if that is what you called the disk.)

(5) Type LOAD RAM.RESTORE

(6) If you use another /RAM driver then edit line 500 to run that driver. Delete line 500 if you have a ramcard that does not need a driver. (You do that by typing 500 <RTN>.)

(7) Type SAVE RAM.RESTORE

(8) Type RENAME RAM.RESTORE,STARTUP

(9) If you have not already done so, run your ram driver, install ProSel on the ram drive, and run BACKUP and backup the RAM volume to floppies. You should also run the MODIFY.BACKUP program to set your desired defaults in the BACKUP and RESTORE programs.

Now try booting this disk from a cold state. Leave the computer off for at least 5 seconds before turning it on again. This should put you into the RESTORE program which will ask you for a source slot. If not, then review these instructions.

It may help to change lines 510 and 520 to give full pathnames:

```
        510 PRINT CHR$(4)"BRUN /RESTORE.RAM/BOOT.PROSEL.R"
        520 PRINT CHR$(4)"-/RESTORE.RAM/RESTORE"
```

When you get it so it successfully gets to the restore program then follow the prompts after removing the boot disk at inserting the first backup disk BACKUP.001. For the destination slot and drive you must use the slot and drive of your RAM driver. For the ProSel RAM driver it is slot 3 drive 1, but other drivers may use drive 2. Remember to run the BACKUP program whenever you add or change files on the RAM disk. This should be put as one of the applications on ProSel.

THE AUTO-BOOT FEATURE

Ordinarily ProSel is designed to be the boot program and after a boot you have to select the first application by hand. It has also been designed, however, to support auto-booting of another application after establishing itself in the "quit code". This can be used for totally hands off rebooting of an application after a power downage. There are many ways of doing this, but I will explain the main one. This is to put your boot application name in the "STARTUP" position inside the PROSEL.SYSTEM file. The "STARTUP" position is at byte 7 of the file (this is at location $2006 when the file is loaded to address $2000). This follows a JMP and two bytes $EE (the startup signature) and a byte $40 (the maximum startup length). The startup (now 0) must be a length byte followed by the pathname. If the pathname is not a full pathname then PROSEL.SYSTEM will automatically append the volume name to it - this is the recommended way because it ensures portability. Note that if PROSEL.SYSTEM fails to find this file or there is some problem loading it, then it will be ignored and you will be sent to the ProSel application listing.

For example, suppose you want the program MY.BOOT to run automatically after booting ProSel. Then assuming MY.BOOT is on the main directory, put the byte 07 (the length of this name) at $2006 and then follow this with the hex bytes representing "MY.BOOT", preferably in "positive ascii". Probably the easiest way to make this patch is with the BLOCK.WARDEN program: Select BLOCK.WARDEN from ProSel. If needed, use the C command to change slot and drive to that with the PROSEL.SYSTEM file to be patched. Type the F command and type PROSEL.SYSTEM as the file to be followed. This will present the first (and only) block of the file. Type E to go to the editor. Type the right arrow 6 times. This should bring up a box asking for the "startup" name. Just type this name (e.g., BASIC.SYSTEM) and press RETURN. This leaves you in edit mode. Then type ESC to get out of edit mode and into R/W mode. Type the W write command, then RETURN to accept the block shown. Type Y to accept the write. Finally, type Q and then Y to quit BLOCK.WARDEN.

THE EXTERNAL PROSEL EDITOR

The program PROSEL.ED is a sophisticated editor which you can run from ProSel to edit application specifications instead of using the more limited internal editor. This editor will let you order the entries as you wish and to group them, unlike the internal editor that automatically puts them in alphabetical order. This editor can be used only with the 80 column version of ProSel!

When the editor is run from ProSel it will read the present application list and show it on the screen in a similar format to that in ProSel. You can move the cursor with the arrow keys. To rearrange the names, place the cursor on the one you want to move, hold down the Open Apple key while moving the name with the arrow keys. Note that moving a name one position swaps it with the one it is moved onto. Thus it is easiest to move it down to an empty area before trying to move it to another column.

To edit an entry, place the cursor on the one to edit and press RTN. This brings up another editing screen which shows the entire application specification. Use the Arrow keys to move the cursor, the TAB key to toggle insert mode, and ESC when done. The RTN key has the same effect here as a down arrow, namely it moves to the next item, or to the first if you are on the startup path.

To enter a new specification just "edit" an empty line. This will allow you to use the "automatic" file specification generator described earlier.

If an "application" title begins with a character less than 64 in ascii then ProSel will ignore it. Thus you can put titles of groups of applications on the screen by using a space followed by the title of the group (eg., " WORD PROCESSORS").

If the title begins with a ^@ then it will be shown in inverse video on the ProSel screen, and the cursor will skip over it. The cursor will also skip over titles beginning with a character less than 64 in ascii. (Eg., space and control characters and empty titles.) The mouse, however, may stop on such entries.

When you quit from the editor (ESC key) you will be asked if you want to save the changes. Use the P option to do that.

The recommended application specification for PROSEL.ED itself is:

```
        Screen title:   Prosel editor
        Prefix:         ?
        Application:    UTIL/PROSEL.ED  (or wherever it is)
        Startup:        PROSEL
```

The startup is not needed, but is recommended. It is how the editor knows where to get the PROSEL file. You can use a full pathname for the startup to edit PROSEL files on another volume.

You can use inverse text and Mousetext in screen titles as follows.

First note that to insert a control character that otherwise would be taken as a command by the editor, you type the ^O key (this means the control-O key and not the two keys ^ and O), and then the control character you want to insert.

To make an entire title show up in inverse text, just put in the ^@ character at the beginning of the title.

To turn on inverse for just a portion of a title, insert a ^O at the point you want inverse to start, and ^N where it should end.

To use mousetext, turn mousetext on by the sequence ^[^O (note that ^[is the ESC key). To turn it back off use ^X^N.

Mousetext characters are given by the regular keys @, A, B, C, ..., Z, [, \,], ^, and _. Note these are uppercase, and that lowercase characters do not work when in mousetext mode.

There are other control characters that give commands to ProSel. These are discussed in the "screen demo" and the "screens" can be examined concerning their use, and also the use of mousetext in general. Here is a brief summary of the command characters:

^B: If this is in any screen title, then ProSel will not display the text usually at the bottom of the screen.

^A^N: This sequence in a screen title defeats inverse that is normally shown when the cursor is on that item. Another ^A turns it back on for the remainder of that title.

^C: This character in any screen title defeats use of the number keys and ESC key while that screen is showing.

MISC. UTILITIES

FIND.FILE

The program FIND.FILE will prompt you for slot and drive and a file name. It will then search through the entire disk for this file. When found, the name of the directory containing the file will be shown and the main data concerning the file will be displayed. At this point you will be asked to hit a key. The RETURN key will cause the search to continue for another file of the same name in another directory. The ESC key aborts the program. The "D" key produces a file dump in hex and ascii. You may select printer output of this. The "T" key "types" the file. In this case control characters other than carriage return will be shown in inverse, or ignored if output is to a printer. If the file is a BAS file then you can also use the "L" key to produce a formatted program listing in which all statements are shown on separate lines and loops and conditionals are indented.

When using any of these modes you can make the listing pause or step by pressing the space bar or you can abort the listing with the ESC key.

If the file is a SYS file then you also have the option of running the program with the "R" key.

You can use * or = anywhere in a file name as a wild card. If the file is not found or if a disk read error (or other problem) occurs then you will be told of this and asked to hit a key, after which you will return to ProSel.

You can also use this to run a BASIC program. To do this, you must tell FIND.FILE the location of your BASIC.SYSTEM file. This is done by specifying the full pathname of the BASIC.SYSTEM file as the startup of FIND.FILE. The simplest way to do this is through the ProSel specification of FIND.FILE. As installed this is already set up pointing to the copy of BASIC.SYSTEM in the main directory. If you move BASIC.SYSTEM to another location, however, you will have to edit the ProSel listing for "File finder". When a BASIC program is run this way, the prefix will be set to the directory containing the program. This directory does not have to contain BASIC.SYSTEM.

FIND.FILE can also find files containing a specified text string. To do this you simply type the string at the "filename or string" prompt, but preceding it with a quotation mark (single or double).

The program will then ask for a directory to search. If you just press RETURN at this point, the entire disk will be searched. If you specify a directory then only that directory, and its subdirectories, will be searched.

You can then specify a filetype to search. For example, if you know what you are looking for is in a file with filetype TXT then typing TXT at this point will limit the search to those files, substantially speeding up the search. If you want to search all files then just press RETURN at this prompt.

You are then asked for a minimum revision date of the files to be searched. If you know the item you want is in a file that was revised since a certain date, then giving that date will speed up the search. If you just press RETURN then dates will be ignored.

Finally, the program asks if it should display the filenames of the files it is currently searching. Answering N will slightly speed up the search if you are doing a search through an entire hard disk.

When the string is found, the program will display the catalog information about the file, and will display the portion of the file surrounding the found text, and highlight the found string. The string will be around the middle of the screen, unless it is very close to the beginning of the file, in which case it will be closer to the top of the screen.

You will be told the "byte offset" of the found string in the file, and asked if you want to continue the search or not, and, if so, whether you want to skip to the next file, or continue with the present one. The byte offset can be used with the Type function to quickly go to the location of the found string, if you do a subsequent file search for the found file.

PIN.PATCHER

This small program will take a Pinpoint (1.2 or 2.0) modified BASIC.SYSTEM file and make a patch which will allow specification of other STARTUP programs when you select BASIC.SYSTEM (as modified by Pinpoint) from ProSel.

MAKE.PPSYS

This program will make a file PPSYS that allows you to call up the Pinpoint accessories directly from ProSel.

Directions (follow very carefully):

0. Install Pinpoint first if not already done.

1. Copy the file MAKE.PPSYS to the directory containing a Pinpoint modified version of BASIC.SYSTEM and set the prefix to that directory.

2. Type CREATE PPSYS,TSYS

4. Type BLOAD MAKE.PPSYS

5. Type CALL 8192

6. Type BSAVE PPSYS,TSYS,A$2000,L$3200

7. DO NOT TRY TO EXECUTE PPSYS - THIS MUST BE DONE FROM PROSEL.

8. Make an application specification for this in ProSel. For example, if the file PPSYS and your Pinpoint accessories are in a subdirectory called PINPOINT then use:

```
Screen title:   Pinpoint utilities
Prefix:         ?PINPOINT=
Application:    PPSYS
Startup:        <none>
```

9. You can then delete the file MAKE.PPSYS at your leisure.

THE PARK.HEADS PROGRAM

This is an "application" that allows you to park the heads on a Sider hard disk, without the inconvenience of going to the DOS 3.3 partition. The program was written by

Jim Thompson
JET AeroWorks
Mesa, Arizona 85203

and he kindly permitted me to put it on the ProSel disk. It is not one of the programs automatically transferred to the hard disk by the installation program because it is Sider specific. If you put the program in the UTIL subdirectory then the application specification for it might read:

```
        Screen title:   Park Heads
        Prefix:         ?
        Pathname:       UTIL/PARK.HEADS
        Startup:        (empty)
```

If you select this "application" then the heads on the Sider will be parked and you can then turn off the computer - or press Open-Apple Control-Reset to reboot. Wait for the message that the heads have been parked to appear on the screen before turning the Apple off.

THE "VT" VERSIONS

On the EXTRAS disk, are versions, CAT.DOCTOR.VT and PROSEL.ED.VT. These are intended to substitute for CAT.DOCTOR and PROSEL.ED on an Apple][+ with a Videx Videoterm 80-column card in slot 3. Apple //e, //c and IIgs users should ignore them. The STARTUP program on the EXTRAS disk will copy these files over the regular versions installed previously.

These versions don't use Mousetext and don't use the up/down arrow keys (which don't exist on the][+). The sort routine in CAT.DOCTOR.VT does use paddle button 0, which is the same as the open-apple key on a //e.

These versions also work on a][+ with Ultraterm in slot 3, but will probably NOT work with many other 80-column cards because there is a bit of hardware specific code in them.

RUNNING DOS 3.3 PROGRAMS DIRECTLY FROM PROSEL

This section relates to hard disks having a DOS 3.3 partition (and NOT for a DOS.MASTER partition). Ignore this section if you have a ProDOS only hard disk.

The short program DOS.HEADER is a DOS 3.3 "header" that is to be appended onto the start of an image of the DOS 3.3 in your DOS partition. (Note that this may not work with some hardware.) Here is how it should be done:

Copy the file DOS.HEADER to wherever you want it on your hard disk. Go to the DOS 3.3 partition. Go to the monitor by CALL -151. Type:

```
        2100<9D00.BFFFM <RTN>
```

Then type BSAVE DOS,A$2100,L$2300.

(DO NOT just BSAVE DOS at its standard address; internal flags are set during a BSAVE that will cause such a copy not to be "clean". This is why the monitor move is used first.)

Next use CONVERT or similar utility to move this file "DOS" to the ProDOS partition. Put it in the directory holding DOS.HEADER and set the prefix to that directory. (You may have to use FID to copy the file onto a DOS 3.3 floppy first.)

From ProDOS BASIC, type:

```
        BLOAD DOS,A$2100
        BLOAD DOS.HEADER
        CREATE DOS.3.3,TSYS
        BSAVE DOS.3.3,TSYS,A$2000,L$2400
```

That's it. Now to run any BASIC DOS 3.3 program directly from ProSel you just have to give an appropriate application specification for the program. Let us suppose that you have put the file DOS.3.3 in the ProDOS subdirectory MISC, for example. Suppose you want to run a program called MYPROGRAM in Volume 5 of the DOS 3.3 partition. Then this specification will do it:

```
        Screen title:   Myprogram
        Prefix:         ?MISC
        Application:    DOS.3.3
        Startup:        MYPROGRAM;5
```

Note the semicolon. The ProSel editor will not permit commas but will take semicolons, so I established this syntax for specifying the volume with the program you want to run. Similarly, you cannot use a space in a STARTUP name, so I provided that the slash "/" be converted to a space by the header. The volume number defaults to 1. A "B" or "T" after the volume number will specify a binary or exec file as the file type of the HELLO program.

DOS 3.3 ON A 3.5" DISK

The programs UNI.HEADER and UNI.FORMAT can be used to partition a 3.5" disk with a DOS 3.3 volume occupying 1/4 of the disk, and ProDOS on the rest. It will boot into ProDOS.

UNI.FORMAT will format a 3.5" disk for this purpose, and can also be used to do a standard 3.5" disk format without the DOS 3.3 partition.

UNI.HEADER is similar to DOS.HEADER and you have to use it to create a version of DOS 3.3 that can be used for these partitioned disks. To do this, first load a "clean" copy of DOS 3.3 into the Apple. It can have some patches, but I recommend against that. Then follow these steps with great care:

```
        CALL -151
        2300<9D00.BFFFM
        3D0G
        BSAVE DOS,A$2300,L$2300
```

Then use CONVERT or a similar utility to transfer the file DOS to ProDOS. Then:

```
        CREATE UNO.DOS,TSYS
        BLOAD DOS,A$2300
        BLOAD UNI.HEADER
        BSAVE UNO.DOS,TSYS,A$2000,L$2600
```

You can now delete the file DOS.

The new file UNO.DOS must be used from ProSel in a similar way to that described in the last section for DOS.3.3. You do not need to specify a volume because the volume created by UNI.FORMAT is volume 1. However, if you use a semicolon followed by B in the startup specification it will BRUN the binary program specified in the startup. Similarly a T after a semicolon will treat the startup as an EXEC text file.

For example if you want to execute FID directly from ProSel you can have this application specification:

```
        Screen title:   Myprogram
        Prefix:         ?UTILITIES
        Application:    UNO.DOS
        Startup:        FID;B
```

Uno-DOS is the most compatible version of DOS 3.3 for Unidisks available. You can use FID on it without modification. You can use it to access floppies as well as UNI.FORMAT created DOS volumes. It uses NO memory overhead except an area inside the formatter routine in DOS; but the INIT command is disabled anyway. It uses standard 16 sector tracks so that many utilities will operate with it. It has, however, 50 tracks. Some programs will not work if they use their own image of RWTS, but these are few. (They include, however, CONVERT. You can use CONVERT to transfer files between ProDOS and a floppy and then use FID to transfer between the floppy and Uno-DOS. The same applies to similar utilities RUNNING UNDER ProDOS.)

Why I chose to partition DOS 3.3 to only 1/4 of the disk:
I feel that ProDOS so greatly outshines DOS 3.3 that the greatest use of this utility will be to run just a few programs that are still only available under DOS 3.3. It is more convenient to do this (much more if you have only one Unidrive) if both ProDOS and Uno-DOS are on the same disk. Also making it 16 sectors per track enhances its compatibility, even though this limits its size to 1/4 of a 3.5" disk. My first version used half a disk and used 32 sector tracks, but I quickly decided the present setup was far superior.

When in the DOS 3.3 partition you can return to ProDOS by typing PR#5 (or whatever slot your 3.5" disk is in).

The ProDOS volume on a partitioned disk looks just like a standard 3.5" disk volume of 1600 blocks. The blocks corresponding to the area used by the DOS 3.3 partition are marked off in the bit map so that they will not be used by ProDOS. In this way, it is possible to use the ProSel COPY program to copy the ENTIRE disk including the DOS 3.3 partition. You MUST NOT use any utility that "recovers unused blocks" because this would free up the blocks on the DOS 3.3 partition and they may be overwritten by ProDOS at some later time.

If you try to CATALOG or otherwise access from Uno-DOS a 3.5" disk that does not have the DOS 3.3 partition you should get an I/O error.

If you have a hard disk that supports DOS 3.3 then you should try to use the DOS 3.3 on the hard disk to attach UNI.HEADER to. This MAY allow access to both the hard disk and Unidisks. Thus, in this case you should not bother with the DOS.HEADER program.

You probably cannot use this on most DOS 3.3 clones because the patch area likely conflicts.

You cannot use Uno-DOS with disks created by other 3.5" disk versions of DOS 3.3 (and vice-versa) because they do not have the same method of translating tracks and sectors into blocks.

Since tracks 1 and 2 are available for data, the Uno-DOS partition has over 50% more data area than a standard floppy.

Important note: Uno-DOS MUST be run FROM ProSel. Do not try to execute it directly.

PASSWORD PROTECTION

The program PASSWORD can be used to give a light password protection to your hard disk. To use it, locate it on the disk. (Installation puts it in the UTIL subdirectory). DO NOT LOCK the file PASSWORD, or you will not be able to change passwords. Then specify UTIL/PASSWORD as the startup inside PROSEL.SYSTEM:

> [To do this, get into BLOCK.WARDEN and type F for follow and then PROSEL.SYSTEM to follow that file. Type E to go to edit mode. There should be two bytes EE after the first three bytes. Type the right arrow key 6 times. At this point a box should appear to ask you for the STARTUP. Just type UTIL/PASSWORD. Then type ESC to go to R/W mode, type W, then RTN, then Y to the question "do you want to write..".

Before doing this, you can test how it works, without danger, by just selecting PASSWORD as an application from ProSel.

The first time you get into PASSWORD it will say "password accepted" (because there is none) and ask you to type a password. REMEMBER WHAT YOU TYPE. When you have completed this you will go right into ProSel. The next time you boot up - or select PROSEL.SYSTEM - you will be asked to type the password. You get three tries. If you type it correctly then you go into ProSel, otherwise, after the three tries, the computer locks up and you will have to hit reset to reboot. Passwords are not case sensitive.

It is easy to change the password. Just hold the open-apple key down when you type the RTN at the end of inputting the current password and the program will prompt for a new one.

You can tell the PASSWORD program to run another program before going to ProSel by putting the name of such a program in the "startup" position inside the PASSWORD file. Do this using BLOCK WARDEN in a similar manner to that described above for putting a startup in the PROSEL.SYSTEM file.

Don't expect too much of this "protection". It will not impede anyone with even moderate knowledge of the machine. But it can be useful if you just want some protection from idle pecking at your keyboard when you are away from your desk for a few minutes. Note that you can bring up the password screen just by selecting PROSEL.SYSTEM from the main directory.

SELECT.SYSTEM

This is an alternative program selector. It is not at all comparable to ProSel in its abilities, but it is entirely memory resident so that it can be used conveniently by those who do not have an appropriate storage device from which to run ProSel. (I.e., our poor floppy bound friends.)

To use this just make it the first ".SYSTEM" file on your boot disk. You can ignore this if you boot using a copy of PRODOS from the ProSel disk, since that has been patched to replace the usual quit code with SELECT.SYSTEM.

When booted or when you quit an application, SELECT.SYSTEM will read all disk drives and display the list of volume names. Use the right and left arrows (only) to select the one you want and then press RTN. The directory from that drive will then be read and the subdirectory file names are displayed as well as the SYS file names.

You can select a subdirectory - then its subdirectories and SYS files will be displayed - and so on.

You can press TAB to switch the highlighted name between the SYS and DIR parts of the screen.

If you press RTN when a SYS file is highlighted, that file will be run with the prefix set to the present directory.

If you press ESC then one level of the prefix is "popped". If that leaves no prefix, then all the drives are polled again just as at the start.

This program must live in very limited space, and hence the error checking is minimal and there are few bells and whistles. Like ProSel, this is fully compatible with the Videx Ultraterm.

SELECT.SYSTEM supports an auto-boot provision that works in exactly the same fashion as that in PROSEL.SYSTEM.

SCAVENGE

This little program SCAVENGE, when executed, will look at all disk devices on line and remove from the ProDOS queue any that do not contain ProDOS disks. The purpose of this is to avoid access of empty drives by programs that poll all devices. If it does not find any drives to remove then it rebuilds the queue. Thus if you run this a second time it will reenable all disk drives.

THE CLOCK.DRV PROGRAM

This is a clock driver for the AE "//c SYSTEM CLOCK". Don't use this if you don't have this hardware. This was written for several reasons. The original clock driver supplied with this had conflicts with the //c mouse. Consequently, it did not work correctly with ProSel and many other programs. Newer versions of it corrected this gross flaw but the year still had to be set by hand. The present driver knows the year and it will work up to 1993. (It will work beyond that by a simple modification.) To use this driver just boot into ProSel and select it as an application. This need only be done once until the computer is turned off again.

ProSel on the Apple IIgs

On the Apple IIgs, ProSel can also run ProDOS-16 application programs (programs of file type $B3, or "S16" files). In order for this to work, the machine must be booted into ProDOS-16 from a boot disk with Apple's START program replaced by the START program on the EXTRAS disk, and with the files PROSEL.SYSTEM and PROSEL on the main directory. Note that installation of ProSel on an existing P16 boot disk automatically sets this up.

To make a ProDOS-16 boot disk, you copy the PRODOS (this is not really Prodos, but the "Loader") file from the SYSTEM.DISK to your boot disk, and also copy the entire SYSTEM directory from the SYSTEM.DISK to the boot disk. (There may be things there you do not need and can delete later.) Copy PROSEL.SYSTEM and PROSEL to the main directory of the boot disk, and copy START from the EXTRAS disk to the SYSTEM directory of the boot disk.

Alternatively you could choose to boot into the "Finder" and select ProSel from there. To do this you just have to make a boot disk (or use the SYSTEM.DISK) and copy the files PROSEL and PROSEL.SYSTEM to it. If you do it this way, quitting from a ProDOS 16 application will send you back to the Finder and not to ProSel.

You can also choose to boot into a P16 program with a quit to ProSel. To do this you must use Block Warden to modify the START file, putting the desired pathname at byte $4B of START. In Block Warden, follow the START file, go to Edit mode, move the cursor down to the $040 line and across till the box appears. Then type the startup name. It is recommended that this name begin with */ which, in P16, stands for the boot volume name.

QUEUEP

QUEUEP (Queue for Prosel - pronounced KEWPIE) is an automatic program queue, allowing hands off sequencing and scheduling of programs. Since most users are not expected to have a need for this, it is not installed by the installation program. To install it, use CAT.DOCTOR to copy the files QUEUEP and QUEUEP.SYSTEM to your main directory.

In appearance, it is very similar to ProSel itself. It also accepts all the ProSel syntax for application specifications. The only difference is that the applications are run in sequence as they appear on the screen. (The sequence goes down the first column then the second column, etc.)

EXPECTATIONS OF SCHEDULED PROGRAMS

A program that is run from QUEUEP is on its own as far as its own operation is concerned. Thus, for automatic scheduling, the programs must execute the QUIT command to ProDOS automatically upon completion. If the program expects user input then it cannot be done hands off. The program must quit before QUEUEP can take over to run the next program.

WHERE TO PUT IT

Since the program works in a very similar manner to ProSel, it must be put in the main volume directory. You must also put its booter program QUEUEP.SYSTEM there, but usually you would not put this before PROSEL.SYSTEM unless you want it to be the boot program.

EDITING THE APPLICATION LIST

To edit the list, you just have to select PROSEL.ED from ProSel with a STARTUP specification of QUEUEP, instead of ProSel. Then use the editor just as you would edit a ProSel listing. Thus, for example, use this specification in ProSel to run the editor to edit QUEUEP:

```
        Screen title:   Queue editor
        Prefix:         ?
        Application:    UTIL/PROSEL.ED
        Startup:        QUEUEP
```

SCHEDULING

If the STARTUP of an application (in the QUEUEP listing) is just a regular startup specification, or is empty, then the application will be executed immediately when its turn comes.

If the STARTUP has a time in the 24 hour format as in

```
        FILENAME:22:30
```

or just

```
        :22:30
```

then this application will be run at 22:30 (10:30 PM). Note that if the last application did not quit before 22:31 then this one will wait till the next day!

If the STARTUP has a time in the format

```
        FILENAME>00:05   or just   >00:05
```

then the program will be executed 0 hours and 5 minutes after the previous program quits. In other works, the ":" designates an absolute time and the ">" designates a relative time.

If the STARTUP (including time, if any) ends with a backslash "\" then any disk error will print its error message, sound a buzzer and wait forever for a keypress. If it does not have this character any error will sound the buzzer for about a minute and then go on to the next application. On a keypress, the next program will be up in the queue. Thus, to prevent its execution it should not be an immediately executed program (it should have a time) so that you can override the queue with the ESC key, or you should keep the closed apple pressed when doing this.

Times on the screen are in 12 hour format but, internally, they are in 24 hour format.

KEYBOARD COMMANDS

After about 5 minutes, the screen will blank to prevent burn in of the monitor. Any key hit at that time will bring the screen back. If the screen is not blank then the right arrow or down arrow keys will skip the presently scheduled application (shown in inverse) and go to the next. If the ESC key is hit, it will go directly to the last application. If the RTN key is hit, it will execute the presently scheduled application immediately. The up or left arrows go back to the previous application.

If the open apple key is held down while an arrow key is pressed then the execution time for the currently selected application is adjusted. (This is temporary. If you move between application specifications with the arrow keys, and open apple not down, then the execution time will be reset.) This feature allows changing the time for the next application without going into the editor. Note that if you pass through the current time while changing the next run time, the selected application will run immediately. You can press the closed apple key (also) to prevent that. Pressing the closed Apple key prevents a program in the queue from running at that particular time, allowing one to pass through the list without the possibility of accidentally having a program execute when this is not desired.

SUGGESTIONS

It is suggested that the last application execute PROSEL.SYSTEM.

For example, use this specification as the last one:

```
        Screen title:   Prosel
        Prefix:         ?
        Application:    PROSEL.SYSTEM
        Startup:        :09:00  (or empty, etc)
```

If you use QUEUEP.SYSTEM instead of PROSEL.SYSTEM in this specification then the whole process recycles. (You would have to cold boot in this case unless one of the programs lets you execute PROSEL.SYSTEM.)

You could have several queues in one listing if, for example, you end each of them with an entry running ProSel. Running ProSel cancels the queue.

When traveling with my //c, I use Queuep instead of Prosel to boot up a file into /RAM. Queuep first runs the ram driver, then the clock driver (both with no time lapse) then runs RESTORE to load the ramcard from a file. This has a 1 minute time delay so that I can choose to skip to the next line which also runs RESTORE to load the ramcard from a different file on the same 3.5" disk. In both cases, RESTORE finishes by executing a version of PROSEL.SYSTEM which is on the restored /RAM volume.

CACHE

CAUTION: At this time the Cache programs are incompatible with Softswitch from Roger Wagner Publ., and MUST NOT be used at the same time as that utility!

The ProCACHE program is a disk cache which uses an area of a large RAM volume to cache data from a hard disk or a 3-1/2" disk to speed disk access. The program will ask what device you wish cached, the default being the device the program was run from. It then asks for the slot and drive of the RAM volume. The default is variable according to what the program finds in your machine. If the program determines that the RAM volume is suitable then it locks out a range of blocks on that volume for use by the cache. It will then monitor blocks read from the cached volume (eg., the hard disk) and place frequently used ones in the ram cache. This can enhance disk operations by a substantial amount. It is compatible with PRODOS 1.1.1 and later but does not work with earlier versions.

The ram volume used must have sufficient CONTIGUOUS room, in excess of 262 blocks, or it will give the error message "Not enough room on the ram volume". The ram volume must also have a name that starts with "RAM". This is used as a safety test to make sure you are not trying to use a volume that is not appropriate to the program. Names like "RAM" or "RAM5" are acceptable. If this is not the case the program will give the error message "Ram volume not found".

The operation of the program is transparent to the user. The program tries to determine potential problems but this cannot be 100% reliable. You must avoid programs which are likely to disable the ram volume or overwrite it with disregard for what is in it. For example you must not copy a disk to the ram volume or restore a file to it, while the ram cache is in effect. With a Ramworks type memory card you must not use the "use extra memory" option in the Volume copy program. To do those things, you must reboot first. Just executing the PRODOS file is sufficient for this. If the RAM driver is rerun then the program attempts to find the space it was using before and tries to use it again. If this is not successful then it may eat up more space from the RAM volume.

The program uses the disk name to determine whether the cached disk has been changed. You should not switch disks with a disk of the same name. To do that, reboot. In case of a large volume (a hard disk with at least 4MB) this check is not made PROVIDED the name of the volume is HARD1. This provides for more efficiency on such volumes. If your hard disk is not named HARD1, you can rename it or use Block Warden to change this default in the CACHE file. It is after the pair of $FFs following the startup name and begins with a length byte. You MUST NOT do this with removable media.

If the program sees that the RAM volume it is using has been disconnected then it aborts with a SYSTEM DEATH message reading INSERT SYSTEM DISK AND RESTART - ERR 0F because this is a condition from which it cannot recover. Unfortunately Appleworks (at least some versions) is misbehaved and disconnects all disk devices in slot 3 and that will produce this error when using some ram drivers, such as the one with ProSel. You can patch your Appleworks to fix this bug as follows: Get into Block.Warden, set the prefix to the directory containing your Appleworks files, type F to follow and specify APLWORKS.SYSTEM. Type E to go to edit mode. Type ^S to search. Type $29 70 C9 30. It should stop with the cursor on the 29 of such a string. Move the cursor to the 70 and type F3 to change the 70 to F3. Move the cursor to the 30 and type B3 to change the 30 to B3. Type ESC to go to R/W mode. Type W to write, then RTN to accept the block shown and Y to accept the write. That is it.

The program will substantially impair 5-1/4" floppy disk access so, if you intend to do much of that, you should reboot. This is due to a large added overhead when accessing floppies and this is unavoidable. The only alternative would be disabling floppy access altogether as earlier versions of CACHE did. The program cannot be used to cache floppy drives and attempting to do so will yield an error message.

You can automate the device selection in the program by putting the four keys for the cached slot/drive and RAM slot/drive after a semicolon following the STARTUP name (if any). An M represents the RETURN key to accept the default. This specification will be ignored if it does not have exactly four characters. For example the startup ";MMMM" accepts all the defaults, and ";7152" acts as if you pressed this sequence of keys when the slot and drive particulars are requested. The startup file itself (if any) will be run following completion of the CACHE setup instead of quitting back to ProSel if a startup name is given. (I do not generally recommend use of a startup with this program.) For example, if the startup specification (in ProSel or in the file itself) is

```
        NEXT.PROGRAM;7152
```

then CACHE will select slot 7 drive 1 for the cached volume, slot 5 drive 2 for the ram volume and then run the program called NEXT.PROGRAM. The specification

```
        ;7152
```

will select those volumes but exit back to ProSel when the cache is set up. The cache setup is very fast and if you use this automated provision you will just see the menu flash on the screen momentarily.

The program CACHE.XL is an alternative version of CACHE which uses twice as much ram area for caching but disables the 5-1/4" floppy drives (reboot or rerun PRODOS to access them). Generally

I think the regular CACHE is to be preferred but if you find too much reloading of files then you might use CACHE.XL instead.

PROSEL.CYCLER and SCREENS

This program lets you keep several different copies of the ProSel file (and the application list it contains). To set this up:

1. Put the PROSEL.CYCLER anywhere on the disk, if not already there. (Install puts it in the UTIL directory.)
2. Use CAT.DOCTOR to copy the file PROSEL (or PROSEL.80, etc., from /FLOPPY) to some temporary directory, and then rename it PROSEL.2.
3. Copy PROSEL.2 to the main directory.
4. It you want another copy, rename PROSEL.2 in the temporary directory to PROSEL.3 and copy that to the main directory.
5. Continue, if desired, up to PROSEL.9.
6. Note that this leaves no PROSEL.1; that is what the current PROSEL file will become if you switch to another copy.
7. Make an application in ProSel which points to the cycler. For example, if the cycler is in the directory UTIL then use

```
    Screen title:   Prosel Cycler
    Prefix:         ?
    Application:    UTIL/PROSEL.CYCLER
    Startup:        (none)
```

8. When this is saved, select this item from ProSel. This will bring up the second copy of ProSel.
9. Go to step 7 for this copy of prosel, and continue until you return to the original copy of prosel.
10. Delete the temporary copy of prosel (see step 2).

Note that the external or internal ProSel editor will always edit the copy of ProSel that is active at that time.

If you hit the Open Apple key when executing the cycler, you will get a menu showing all options and can select the version of ProSel you want to go to next, instead of cycling through all versions. This is convenient if you use more than 3 or 4 versions. There is a small editor CYCLER.EDITOR that can be used to change the names shown on the screen in this menu mode of the cycler, and it also lets you reverse the meaning of the Open Apple key, so that the menu becomes the default, and lets you select 40 or 80 column mode for the cycler display.

If you call the cycler from ProSel with a specification in the menu giving a STARTUP name consisting of a digit (1-9) then the cycler will bypass the menu and will bring up the version of the PROSEL file corresponding to this digit. For this to work, you must use the CYCLER.EDITOR to make the MENU mode the default.

ProSel also supports extra "screens". There is a demo of these on the back of the EXTRAS disk (or in the SCREENS directory), and the demo includes some documentation on their usage.

A "screen" is just a ProSel application list that is saved by the external editor into a special type of SYS file that can be executed by ProSel. It is in the nature of a screen, however, that it leaves ProSel in memory and just replaces the application list by the one in the screen. This provides a very flexible extension of ProSel which is virtually unlimited since there is no limitation on the number of screens that can be used nor on the location of these files on the disk.

There is really nothing much to know about screens. They can be loaded into the external ProSel editor and edited and saved back again. There is nothing different about this than editing the standard ProSel listing, except that you use the "Save screen" function in the ProSel editor rather than the "Save new ProSel" function. Screens are "selected" just like any other executable program.

Although there is nothing different in screens than in a regular ProSel listing, the demo screens show how one can make a screen look quite different from a standard application list. (The same things can be done with ANY ProSel listing, however.)

There is a command (a ^B anywhere in any application specification screen title) that will cause the usual bottom of the ProSel screen to be hidden from view and enabling regular titles going all the way to the bottom of the screen. To put titles below the usual bottom boundary, you use the ProSel external editor, place the cursor on the lowest line it will go to, and hold the Closed Apple key down while pressing the down arrow. This will "release the lock" and permit the cursor to go down to any line below the usual bottom.

There is another command (^C in any title) that disables the usual ESC and number key usage in ProSel. This is so you can force selection of some item on the screen and disallow exit through the "back door". Also, note the discussion of inverse and/or mousetext titles in the section on the external editor.

For further ideas about how "screens" can be used, go through the screen demo. Simply access the disk or SCREEN or directory SCREENS and select the item called SCREEN.DEMO. That, in turn will allow you to access further screens which talk about this feature. Also, to see how some things are done, use the editor to look at the actual application specifications in any of the example screens. It is not recommended that you try this feature until you are thoroughly familiar with other aspects of ProSel.

BACKUP, RESTORE and RECOVER

These programs are designed to do one thing, provide fast full volume backups for hard disks using ProDOS, and do it extremely well. These programs require a 128K Apple //e. They also use "mousetext", but this is just cosmetic. Versions of these programs that work on the][+ with 64k are on the /EXTRAS disk with a ".PLUS" extension.

BACKUP

This program is used to back up a hard disk volume onto floppies. It will work with 5-1/4 inch floppies, 3-1/2 inch floppies, and in fact with any ProDOS formatted disk media at all. (The program does require the backup media to have less than 4096 blocks. This is to guard against accidental "backup" TO the hard disk.)

The backup disks must be ProDOS formatted before using the program, UNLESS the drive is an Apple 3.5 (NOT Unidisk 3.5) in which case an unformatted disk will be automatically formatted with a 2:1 "interleave". It is most efficient to give them volume names BACKUP.001, BACKUP.002, etc. If the disks you use do not have these names, you will be asked if you want to destroy the existing volume, and the program will rename the backup disk if you agree to this.

The backup program will use both drives alternately for the floppies unless you specify only one is to be used (see below).

When you enter the program, you specify the Original slot and drive which should be that of the hard disk volume you wish to backup. Later the program will read the volume name and display it (here shown as /MYVOL). You also specify the destination slot and drive. The drive is the first one used by the program for the floppy, but unless you use the single drive option (below) the drives will alternate. Finally you are asked if you want to use two drives for the backup disks (default = Yes).

The backup procedure will start as soon as you answer this question. The first backup disk should be in the specified drive. While this first drive is being operated, insert the second disk in the alternate drive. When the first disk is done and the second is being used, insert the third and so on. The total number of disks that are going to be used is shown after "Number needed".

If the disk you have inserted is not named properly (BACKUP.xxx) then the process will stop and you will be asked if you want to destroy that volume. You can choose to abort the backup process by hitting "N" at this time. If you have inserted the wrong disk, replace it and hit "Y". The disk will be renamed and the backup process will then continue. Note that "destination drive" shows the drive currently expecting the backup disk and the "BACKUP.xxx" name shown is that of the volume expected.

If the program does not recognize the backup disk as a ProDOS disk or if there is no disk in the drive then the process will pause and you will be asked to "Reinsert/replace backup disk". Possibly the disk was not well centered. Try it again, or try a replacement disk and type C for continue. The escape key aborts the backup process.

If the program is unable to read some block from the hard disk or is unable to write a block to the backup disk, it will retry four times and then inform you of the problem. The block that is causing the problem is shown at the bottom of the screen in inverse. You can choose to ignore this error and continue the backup process. (Make a note of the problem block if you do this. That block will not be transferred correctly.) If the problem is a write error, then you should abort the process by hitting the ESC key and do it over, possibly with some other backup disks. Don't take a chance on faulty media.

To insure maximum reliability we recommend formatting the backup disks on the same drive to be used by the backup program. Ordinarily this means formatting odd numbered disks on drive 1 and even numbered ones on drive 2. The program will automatically format unformatted disks if they are in an Apple 3.5 drive (not a Unidisk 3.5). Please note that it takes a few seconds for the program to recognize this condition, so be patient - actually it is the ROM firmware that is slow here, so there is not much that can be done about it.

There are several user modifiable parameters such as original and destination slots. You should run the MODIFY.BACKUP program (a BASIC program) to change these defaults.

One of the parameters for BACKUP is to clear the "backup bits" of files. Ordinarily you should NOT use this option. It is designed to be used by people who ALSO use another backup program to do daily incremental backups, while using ProSel's faster backup to do weekly full volume backups. If you only use the ProSel backup then the backup bits are of no concern to you.

Another parameter is whether you want BACKUP to do a verify after writing each backup disk. Ordinarily, I don't think this is worth the extra time it takes. However, if you do use it, and a read error occurs during a verify of a backup disk, then you will be given the option of replacing that disk with a good disk and continuing the backup process.

When the BACKUP/RESTORE programs are expecting you to insert a disk and IF that drive is an Apple 3.5 (NOT Unidisk 3.5) then the backup or restore will restart automatically as soon as the disk is inserted and you do not have to press the C(ontinue) key as you would with other backup media. Please note that if the disk is not formatted then it will take a few seconds before formatting of the backup disk starts. (Also note that formatting is supported ONLY for Apple 3.5 drives, NOT for Unidisk 3.5's or 5.25 floppies.)

RESTORE

This program is used to restore the hard disk volume to its condition when the backup was made. CAUTION: Any additions since the backup will be lost. The primary purpose of RESTORE is recovery from a disaster that has destroyed the hard disk volume. (If the hard disk needs reformatting, do that before using RESTORE.) Obviously you should have a copy of the RESTORE program on a floppy disk.

When you enter the program you specify the original slot and drive, which is the slot/drive of the backup floppies and defaults to slot 6, drive 1. The destination slot/drive is that of the hard disk. You MUST specify the correct slot and drive for the hard disk. On the Sider hard disk, drive 1 corresponds to the volume /HARD1 and drive 2 to /HARD2. If you backup from drive 1 (HARD1) and then try to restore to drive 2 (HARD2), you will get an error message reading "Incorrect volume size/name". If the volume size is correct (see below) then you will see the expected volume name on the screen after "Destination slot" (here it would be /HARD1). If you still want to backup to that volume, you will have to exit the program, rename the destination volume and rerun RESTORE. This is a safety feature to prevent accidentally overwriting the wrong volume.

The RESTORE program demands the backup disks to have the correct names. If one comes along with the wrong name, the program will pause and ask you to insert the correct disk. (For this type of error the C[ontinue] key and R[etry] key are equivalent.) The disk name "BACKUP.xxx" is displayed after "Original slot" and the active drive is shown after "Original drive". When you change that disk and press C or R the program will look at the replaced disk and make sure it is now correct. (If not, you get the message again.) Hitting ESC at this time will abort the restoration process and will leave your hard disk with incorrect data. You can, of course, rerun RESTORE from floppy and try again.

You must restore to a volume of the same size as the one backed up. The program will not accept a different size. If you have to reformat the hard disk, make sure you create the same size volume. If you want to change volume sizes, you will have to use another backup utility to do it. This is a consequence of the way this program works and is partly responsible for its speed. [The current version will allow restoration to a larger volume (you are asked if it is ok) as long as the number of "bit map blocks" is the same and the old and new numbers of blocks are both multiples of eight. A new "bit map block" is needed for every 2MB of disk space, so this allows a moderate increase in the size of the volume. If you use RESTORE on a /RAM volume you must be sure to use the same RAM driver for restoration as for the backup, and an increase in volume size will work only for the driver supplied and for the Ramworks and Checkmate RAM drivers.]

If a read or write error occurs while attempting a restore, you will have the option of continuing, retrying or aborting. Selecting "retry" will attempt to read or write the problem block again. (You might try reinserting the disk before hitting the "R" key.) The continue option will skip the problem block (which will leave the destination block with incorrect data). If it is a read error, you should note the number shown in inverse under the beginning of "Restoring block" at the bottom of the screen. This is the block on the hard disk which will have incorrect information written to it. This block will contain the message "BAD RESTORE" written into the first few bytes. In a pinch you may be able to fix this block by hand. The block on the backup disk that could not be read is shown in inverse under "Source block" at the bottom of the screen.

BACKING UP AND RESTORING TO AND FROM A FILE

The programs also have the ability to backup a volume to a FILE on a large disk device. This is intended for saving the contents of a /RAM disk. [Note that other methods to use BACKUP to do this are discussed elsewhere.] To get into this mode, just type a 0 for the destination slot in BACKUP or for the source slot in RESTORE. You will then be prompted for pathname of the file to be used. If the file exists (of the proper file type - which is $F6) then it will be overwritten unless it is locked. If it does not exist, it will be created. Note that the entire file must be contained on the destination volume.

If you get into this mode by mistake, type ESC at the pathname prompt.

This ability can be made into an application selectable from ProSel by making application specifications with BACKUP and RESTORE as the applications and the file to be used as the backup file as the "startup". In the case of BACKUP, you also have to give the name of the volume to be backed up to the MODIFY.BACKUP program. If done correctly, then selecting these applications will automatically (no further user input) backup/restore the ram disk to/from the file.

AUTOMATIC BOOTING AND LOADING OF A RAM VOLUME FROM A FILE

Provision has been made for a totally hands off loading of a RAM volume upon boot. This takes a little careful work to set up, but once done, it is automatic and it is very fast.

These are the steps to follow:

1. Install ProSel on the bootup disk.

2. Use BLOCK.WARDEN to place a STARTUP name in PROSEL.SYSTEM. If you have the Apple memory card then put in UTIL/RESTORE, and skip step 3. Otherwise put in UTIL/RAM.DRV and use CAT.DOCTOR to copy RAM.DRV to the directory UTIL on your bootup disk.

3. [Skip if you have an Apple memory card or clone like Ramfactor.] Use the CONFIG.RAMDRV program to place a STARTUP name of UTIL/RESTORE in the RAM.DRV program.

4. Use MODIFY.BACKUP to specify the name of the FILE to be restored to the RAM volume in the STARTUP position of RESTORE. You might call such a file RAMFILE for example, or MISC/RAMFILE if it is in the subdirectory MISC.

5. If you want RESTORE to run some SYS program (eg. BASIC.SYSTEM) instead of returning directly to ProSel after loading the RAM volume, then use MODIFY.BACKUP to place the name of the desired SYS program in the RESTORE program. [Even if this is done, it will be ignored if there is no specification in the STARTUP position (item 4). This makes it possible to use the startup specification in ProSel to run this version of BACKUP while still being able to use the same copy of BACKUP (with nothing in the startup position) for its primary purpose of backing up a hard disk.]

6. Load the files you want into the RAM volume and use BACKUP to backup the volume to the file you named in step 4.

If you have followed this procedure correctly then the next time you boot, the RAM volume should be loaded automatically from the file you created. If not, then you missed some detail. Check the names you have given for various files and make sure the files are really where they will be looked for. This provision will just ignore file specifications if the files are not found. There may be no error messages.

Remember that these instructions apply to loading the /RAM volume from a FILE on a hard disk or 800K disk. To load the /RAM volume from floppies requires a different procedure which was described in the section "USING PROSEL ON A /RAM VOLUME".

AUTOMATIC BACKUP OF A RAM VOLUME

BACKUP can also be instructed to automatically backup a RAM volume to a file and then (optionally) run another application such as PARK.HEADS. Thus a single selection from ProSel can automatically backup your RAM volume, and then park the heads of the hard disk.

To force this mode, the file name of the backup file should be placed in the startup position. [It is best to let ProSel do this via an application specification. That way BACKUP can be used in other ways without making a completely separate version. If the startup position is empty then the rest of the operation described here will be ignored by the BACKUP program.]

Then the program name to be run after BACKUP is finished should be placed in the BACKUP program by selecting that option when running MODIFY.BACKUP.

Then the volume name of the volume to be backed up (eg, /RAM) must be specified when running MODIFY.BACKUP, or this mode will be IGNORED.

USING BACKUP DISKS IN SLOT 5 AND SLOT 2

As set up, the Backup/Restore programs alternate backup disks in a specified slot and both drives 1 and 2. However, the programs are designed in such a way that you can, instead, have it (for example) alternate between slot 5, drive 1 and slot 2, drive 1 (a common configuration for 3.5" disks on a GS). To enable this, just run the MODIFY.BACKUP program to modify both BACKUP and RESTORE. Tell the program the first backup disk location (eg., slot 5, drive 1) and the second (eg., slot 2, drive 1). Use the same slot/drive for both if you want single drive operation without having to answer the drive question when running BACKUP or RESTORE.

RAM.DRIVE

The program RAM.DRIVE is a /RAM volume driver for the Checkmate Technology Multiram and Applied Engineering Ramworks cards. The driver was developed for compatibility with the "main memory" option of Merlin-pro, which is recommended for use with speed up cards. The driver is also very space efficient and leaves many more free blocks on the ram disk than do other drivers. However if another /RAM driver better suits your needs, then you don't have to use this one.

The program called RAM.DRV is an identical driver but it is written as a SYS program (exiting through the QUIT protocol) and thus can be used as a ProSel startup program, etc.

This /RAM driver does not use any of the standard auxiliary 64k space, so it should be compatible with all programs using auxiliary memory. There are no conflicts with double hires graphics.

The /RAM volume will correspond to slot 3 drive 1 rather than drive 2. This is done to avoid problems with some programs that do not handle slot 3, drive 2 devices intelligently. Ordinarily running the ram drive programs will not re-format /RAM if certain ID bytes indicate the /RAM volume is still there. This check is not always reliable, and you can force re-format by holding down the closed Apple key when running the ram drive programs.

The driver has a "bank lockout" that can reserve ram banks for other usage. This can be set by running the CONFIG.RAMDRV program, which also lets you change the slot/drive used by the ram driver and other user selectable parameters.

The files RAM.DRIVE.16 and RAM.DRV.16 are compatible drivers that can be used if (and only if) you have a 65802/816 processor. If you try to use them without this hardware, they will tell you. Their advantage is speed, particularly if you have a 65816.

RAM.DRV.GS is a ram driver for the Apple IIgs which can supplant or supplement /RAM5. If you hold down the option key when running it then it will REMOVE itself. Caution: everything that was in /RAM will be lost forever. It is twice as fast as /RAM5.

If you hold down the open Apple key when running RAM.DRV.GS then it will present a configuration menu. You can resize the ram drive to have any size that is a multiple of 64K, up to 2048K. This does not check how much memory you have available, so be reasonable. (When the driver is run it will abort with a memory manager error if it cannot get enough memory.) Then you can select the slot and drive to map the driver to (default 3,1). When you have specified these parameters, it will resave itself. (Don't lock the file or this will not work.)

RAM.DRV.GS works with ProDOS-8 only. If you run a P16 program then the driver will release its memory to the system. CAUTION: this means that anything in /RAM will be lost forever.

The strengths of this alternative to /RAM5 are also its weaknesses. For example, the driver is easily discarded so that you can run P8 programs which require much expansion memory, such as COPY.GS and Beach.Comber.32. This also means, however, that it is easy to lose data in the ram volume. Care must be exercised.

RAMDRV.HEADER is a "header" that can be attached to the Multiram or Ramworks /RAM drivers so that they can be used in the same manner as described for the RAM.DRV, for automatic restoration of the /RAM volume on bootup.

To use it, get out your /RAM driver of choice. Notice its length in the catalog and add 256 to it - this will be the length of the modified driver. BLOAD your /RAM driver to address $2100. Then BLOAD RAMDRV.HEADER. Then CREATE RAMDRV,TSYS (or whatever name you want) and BSAVE RAMDRV,TSYS,A$2000,L? where you put in the length calculated above. [NOTE that this final file MUST be a SYS file.]

Follow the directions above for using this with the BACKUP and RESTORE programs.

[CAUTION: You must never backup using one ram driver and try restoring to another ram driver. This WILL NOT WORK and may cause other damage!]

REMARKS

You will find that BACKUP is so fast that it will do a full volume backup in less time than other backup programs will do an incremental one. The program also does not manipulate the "backup bits" unless it is altered to do so (see above). This has no real purpose if you do only full volume backups. In fact this program does not manipulate the data on the disk in any way.
You will find that a restored volume is identical to what it was when the backup was made, right down to the create dates on your files.

You can test these programs using only scratch floppies to see how they work before you commit yourself to using them on hard disk. To do this just specify one of the floppy drives as the original and the other floppy drive as the destination.

RECOVER

This is a utility that allows you to recover a file from the backup disks created by the BACKUP program. This can be very useful if you should ever be in a position where your hard disk is down and files on it inaccessible.

To use this you must have two working disks one of which must be of the type used for making the backups. One of these drives will be used for reading the backup disks and the other will be used to write the recovered file.

The program will ask for the slot and drive that you will use for the backup disks.

It then asks for the pathname of the file you want to recover. This name MUST NOT include the volume name. Thus if you should want to recover /HARD1/MYDIR/MYPROG, what you type here must be MYDIR/MYPROG.

Then you must specify the FULL pathname of the filename you want the recovered data written to. If the file already exists you will be asked if it should be deleted. If you answer YES and the file is locked then the program will abort.

Next you will be asked to insert BACKUP.001 and press a key. It may ask for many more of the backup disks. Be prepared for a lot of swapping if the file is large or high up in the hard disk.

If the operation is successful the program will ask if you want to retrieve another file. If recovery is not successful then some error message will also appear.

RECOVER cannot be used to recover files from the file created by the "backup to a file" option.

DOC for COPY

This is a volume copy program for ProDOS disks which features:

1. It is fast, especially on partially full disks.
2. It tries hard to read and write and allows skipping over errors.
3. It can be used to copy one hard disk volume to another of the same (or slightly larger) size.
4. It allows single drive or dual drive copies.
5. It will use the extra memory in a ram card when doing a single drive copy (unless you tell it not to do so). In all likelihood it will take only one pass even for 800K disks if you have a large ram card. Even if you do not have such a card the program will use most of the extra 64K in a //e or //c.
6. If only one pass is required then you have the option of copying to subsequent disks without reading again from the original.

You are asked for slot and drive for the original and duplicate disks. These will be used for all subsequent copies, so you don't have to keep repeating these specifications. The program checks to see if these correspond to mounted devices and, if not, returns to the start. The program then checks to see if the duplicate volume has the same number of blocks and gives an error message if not. (Thus you can copy only between the same type devices. It is impossible to accidently copy from a floppy to a hard disk.)

COPY displays the names of the original and the duplicate and you are asked if you really want to destroy the latter. If not, it will quit. If so the copy will proceed. The creation dates of both disks are also displayed unless no clock was present when the volumes were formatted. The block currently being read or written is displayed at the bottom of the screen in inverse. Only blocks listed as used in the bit map are copied.

If a read or write error occurs the program retries four times If still unable to do the read or write, the program will tell you the error and ask if you want to continue or not. If not it will exit through Quit. If you want to continue anyway (and try to fix the bad block by other means) you should make a note of the block on which the error occurred and then type C for continue.

When the copy is done you will be asked if you want to do another copy. The same original and destination drives are used.

If the volume is /HARD1, /HARD2, etc, the copy will be named /COPY1, /COPY2, etc., and vice-versa. This prevents duplication of volume names when using one Sider to back up another.

Affect on the /RAM volume: If you don't use the single drive, use extra memory option then there is no effect. If you only have the standard /RAM volume, then use of the extra memory option will cause all data in the /RAM drive to be destroyed, and /RAM will be reformatted on exit. Other /RAM drives will also lose their data, and will be reformatted provided they accept a format request. (The supplied RAM.DRIVE does not accept formatting - there is no room in the driver - and so it is left disconnected and RAM.DRIVE will have to be rerun after the exit from COPY.)

Use on 40 track floppies: The program will work on 40 track floppies (or 39 track, etc) but must be used under a ProDOS that has been modified to support 40 tracks. If you try to copy a 40 track floppy to a 35 track ProDOS formatted floppy then the program will automatically format the destination disk for 40 tracks (or however many the original has). This ASSUMES that the destination disk really has the correct number of blocks which indicates 35 tracks and not 40. If you previously used another copy program, copying only the first 35 tracks, then the destination disk will indicate 40 tracks to the COPY program and automatic formatting will not be done. Then when COPY reaches track 36 it will fail. Thus, when in doubt in such situations, press the Open-Apple key (when indicated on the screen) to force formatting of the destination disk. Then the only problem likely to be met is that the disk drive cannot access the extra tracks.

The program allows copying a volume to one of a larger size, provided that has been formatted. One reason for this is to allow copying of an 800k disk using the Apple memory card, by first copying to the RAM disk and then to another (formatted) 800k disk. For this to work you must have 1 Meg in the Apple memory card. You are asked if you really want to do this.
You are also asked (if the program determines that the situation is one it can handle) whether you want the size increased to the size of the destination disk. You would want this if you are copying a floppy to a 3.5" disk, but not if you are copying a floppy to the Apple memory card with a view to copying that back out to other floppies. If you do not select this option then the destination volume will have the same size as the original.
You should reformat the /RAM drive (turn the computer off and on) after using it to copy disks in this way.

CAUTION: The RAM disk method also works with the ProSel ram driver and the AE ProDrive version 5.3 at least. You should NOT use this method with other RAM drivers.

On the IIgs the ProSel installation program will install the version of COPY called COPY.GS and will rename it to COPY. (The renaming is to support the default ProSel menu. You can rename it back and edit its ProSel specification if you like.) COPY.GS looks for free memory and uses whatever it can find. It displays the amount of memory it has found before the copy process begins. COPY.GS also has a "mass copy" feature in which it will read a disk to memory and copy it out to alternating drives without prompting. The drives used default to source S5,D1 and destination S2,D1 alternating with S5,D1. To change these defaults you use a "startup" consisting of a semicolon, slot/drive of source, slot/drive of the first destination and slot/drive of the second destination.
For example ;515152 gives source S5,D1, and destinations S5,D1 and S5,D2. (In cases where the first destination is the same as the source, the program starts writing to the SECOND destination.) If both destinations are the same, mass copy mode will be defeated. In COPY.GS (only), if the destination disk needs formatting and it is a 3.5" disk, then you will be asked for an "interleave" value between 1 and 9. This will come up with the cursor on the recommended default of 2:1 if the drive is a Apple 3.5 and 4:1 if it is a Unidrive 3.5. Ordinarily you should just press RETURN to accept that default, but you may wish to experiment with other values. In mass copy mode, this question is asked only for the first format.

INFO.DESK

This program will print the entire tree structure of a selected volume. The output can be sent to the 80 column screen, to a printer or to a disk file.

There are five main parts to the program that give you different types of documentation of the files on your disk:

1. Catalog.

   This mode prints a tree structured catalog of the entire volume. Subdirectory contents are indented two spaces. All the usual catalog information is printed except the time of day and the access (locked) status. Access status is printed if the line length is set to 90 or more.

2. Block usage by files.

   This mode tells you just what blocks on the disk are used by each file. The printout gives the number of blocks in each file (as in mode 1). Then comes the Index field. This contains the number of the index block. This field is empty for directory files and for seedling files (which have no index block).

   Next come the actual data blocks. If two numbers in this list are separated by periods then they represent a range of blocks all belonging to that file.

   In case of a tree file (length > $20000 = 131072) the first entry in the Index column is the master key block indicated by " <== (Master index block)" next to the block number. After this, on subsequent lines, are the index blocks pointed to by the master index block, and their associated data blocks.

3. File usage by blocks.

   This mode is the "reverse" of mode 2. It shows which files belong to the blocks on the disk. Most of the disk operations are done prior to any printing, so be patient. The printout consists of ranges of blocks in numerical order followed by the file names (the full pathname less the volume name) which use these blocks. This routine must build two large tables and there may not be sufficient room to handle very large volumes. In that case a partial table will be printed. You can tell whether this happened by comparing the number of files with that given by one of the other modes.

4. Bit map.

   This mode prints the volume bit map. Used blocks are shown with "x" and free ones with ".". (You can change these.)

5. Tree structure.

   This mode prints a tree diagram of the volume.

At any time during printout the space bar will stop, then step, the process, and ESC will abort it. When printing to the screen is finished, the program waits for a keypress before clearing the screen and asking if you want to do another.

CONFIGURATION OF INFO.DESK

The program has been designed so that certain parameters are easy to modify. Most people will have no reason to attempt this. The configuration area starts at relative byte 3 in the file. The first part of the data area consists of three segments of 16 bytes each. These three segments are used by output to, respectively, a printer, a disk file, and the 80-column screen. The 16 bytes in each segment have this meaning (byte addresses are given for the first, printer, table with present contents shown):

$03:	01	Slot number for output (0 for disk file)
$04:	3C (dec. 60)	Number of lines/page
$05:	50 (dec. 80)	Line length
$06:	DF ("_")	Fill character after file names
$07:	F8 ("x̄")	Used block indicator in bit map
$08:	AE (".")	Free block indicator in bit map
$09:	7F	Mask for file output (use FF for neg ascii)
$0A:	00 00 00 00 00 00 00 00	Printer (etc) init string, 7 bytes maximum, null terminated.
$12:	05	Left margin

If the number of lines per page parameter is zero then the headers (except the first one) will not be printed. This is mainly useful for the print to disk option.

Following the three tables (at byte $33) is a table of the file types used by the catalog. The format of an entry is the type name in negative ascii followed by the type itself, for example, "BIN",06. Towards the end of the list (currently at byte $87) there are repeated designations "NON",00. Any or all of these may be changed to provide a recognized name for any file type. If a file type is not in the list it is printed in its hex form.

If you choose the option to output the data to a disk file the file will be placed in the directory to which the prefix was set upon entry. (You can use ProSel to set this to any directory to which you wish to have the file sent.) The file name used when the Catalog option (#1) is in effect is "CAT." followed by the volume name. (If this is longer than 15 characters then it is truncated.) If this file already exists it will be overwritten unless it is locked. If it is locked the program will abort. Similarly, for the options 2-5 the file name is "LOC.", "BLK.", "BIT.", or "TRE." followed by the volume name.

CAT.DOCTOR

This program is a filing and directory manipulation utility. It requires a //c or a //e with the enhancement (mousetext) ROMS. Its main function is to do batch copying, locking, unlocking, and deleting of files from specified directories. Most of the routines are limited to directories containing at most 204 entries, but more than 110 will cause truncation of file names on the display.

When a pathname is requested, you are shown an existing one. You can accept that name by pressing RETURN - you do not have to copy over it.

Throughout, pathnames must be given in their FULL form. The TAB key moves the cursor past the next "/", or to the end of the name. Open-Apple TAB moves the cursor back to the preceding "/". In some routines you can select the directory pathname by a tree search through all directories on a volume. When applicable this is indicated on the screen, and the ? key starts the process. If you type the ? as first character then disk devices will be scanned and the volumes on line will be displayed for selection for the tree search. If you type, or move the cursor over, the volume name and THEN type ?, the device search will not be done and the tree search will be done on that volume. The program maintains two default pathnames. These can be switched by pressing ^X at the first character of the first default name. Selection of directory names supports some line editing features: The ^D key deletes the character under the cursor, DEL deletes the one to the left of the cursor. If the open apple key is held down while another legal key is pressed then an insert is done instead of the normal overstrike.

When indicated, and when the cursor is on the first character during directory name input, pressing the closed apple key will bring up a request for slot and drive, with the currently logged device as the default. The specified device will then be accessed through the tree search.

COPY FILES

This is a batch file copier. It retains both create dates and modification dates of files. It is compatible with all types of files including sparse files and subdirectory files. In the case of subdirectory files, however, you will be asked if you want the files INSIDE to be copied - otherwise the new directory is CREATED rather than copied, if it does not already exist. If "prompting" is OFF then the answer to this question will be taken as "Yes".

You specify the source directory and the destination directory. Then a list of the files in the source directory is displayed. Move with the cursor keys to highlight the files you want copied and select them (or deselect them) by pressing the space bar. When ready to copy, press RETURN. If you decide not to copy any files press ESCAPE.

You can select/deselect ALL files by pressing ^A. (Actually, this "toggles" the flags so that previous selections will be deselected and vice-versa.) If "prompting" (see the main menu) is OFF then files will be copied whether or not they exist or are locked on the destination directory. If prompting is ON then you will be prompted if a file of the same name exists on the destination directory, and you will be prompted again if it is locked. During prompting the file name in question will blink. Files that are being copied or that have been copied will show in inverse. When the copying is completed you will be asked if you want to copy the same files to another disk with the same directory name. This allows you to back up files to several disks without going through the process of selection over and over. Also, if you hit DELETE (^D for the VT version) at this point you will be sent to the delete routine at the point of file selection with the same files selected. If you then hit RETURN, the original files will be deleted. (Note: This question is not asked if files inside subdirectories have been copied, because needed data is no longer in memory.)

When files have been selected, you can type ^C (for "changed" files), and the program will then automatically copy only those files among the selected files whose modification date/time is later than that of the same files on the destination directory, and all files that do not exist on the destination directory. The copying will start immediately without need of hitting the RTN key. Note that if you do not select any files before using ^C then none will be copied. To copy ALL changed files, use ^A then ^C. A ^E (exists) command works the same as ^C but copies only the modified files that already exist on the destination directory. The DEL key at this point will mark for deletion from the source directory all selected files that exist on the "destination" directory with at least as late a revision date. (Note that although this is in the COPY routine, this use of DEL does not copy anything.) In contrast to the ^C and ^E, DEL does not take immediate action and leaves you in select mode, but remember this is now selection for DELETION.

TYPE FILES

This option will type text files (or any files) to the screen. You can select any number of files for display. The catalog information for the file is shown just previous to the file display. Most keys will stop the display or restart it.
The ESC key aborts the display and returns to the main menu.
The RETURN key aborts listing for the current file and proceeds to the next selection if any. At the end of each file display, the program will stop for a key press before continuing with the next file, or to the main menu in the case of the last file.
The S key slows the display.

LOCK FILES, UNLOCK FILES and DELETE FILES

This operates much as does COPY FILES, but only those files that are appropriate to the operation are displayed. (Eg., for LOCK only the unlocked files are displayed.) If you try to delete a directory file it will be done only if that directory is empty.

RENAME A FILE or VOLUME NAME

After selection, file names are displayed at the bottom of the screen and can be changed from the keyboard. In order to change the VOLUME name, just select some file for renaming and, when presented with that file, move the cursor back and change the volume name. Press RTN when the cursor is at the end of the changed volume name. This will change the volume name and then present the file again with the new volume name. Just press ESC if you don't want to change the file name.

EXHUME FILES

This function lets you revive deleted files, provided the file has not been overwritten. This works ONLY on PRODOS 1.4 or later. You should also realize that it is a dangerous routine. If a problem is detected then you will get a "file damaged" error message. It might be wise to use MR.FIXIT to check for any problems if anything strange happens with this function.

> DANGER: You MUST NOT use the EXHUME function with files that were deleted under a PRODOS prior to 1.4. The PRODOS file on the ProSel disk is 1.4 with a patch to correct a bug in PRODOS 1.4. You should copy this file to your boot disks if you think you will use the EXHUME function.

Remark: If the EXHUME function is used while running under a PRODOS prior to 1.3 then you will get the "file damaged" message, even though the file may be recoverable under PRODOS 1.4.

VERIFY FILES

This will read specified files to test for bad blocks. If an error occurs then the block will be shown at the bottom of the screen and the program will wait for a keypress. The RTN key will continue from this point, DEL will continue from the next selected file, and ESC will abort. Some disk errors on blocks not really inside the file (finding the file for example) may cause errors aborting the routine. Directory files are not themselves verified, but the files INSIDE directories will be verified if you ask for that to be done (automatic if prompts are off). [You could use BLOCK.WARDEN to read through a DIR file if you are having trouble with one.]

SORT DIRECTORY

This powerful directory sorter asks for a directory name and then displays the names in that directory. Some instructions appear at the bottom of the screen. It accepts the following commands:

```
        A - sort alphabetically
        C - sort by creation date
        M - sort by modification date
        T - sort by file type
        P - sort by file type and alphabetically within a type
        R - reverse present file order
```

You can also use the Open-Apple key in conjunction with a cursor key to move file names around by hand.

When you are done, hit RETURN. (Or hit ESCAPE to abort.) When you hit RETURN you will still be given a chance to abort before the sorted directory is written to disk. WARNING: If you sort the main directory, be sure you leave PROSEL.SYSTEM as the first ".SYSTEM" file. Otherwise ProSel will not boot correctly.

WIPE A VOLUME

This will erase all files on the designated volume. It gives you a chance to change your mind before the action is taken. This is like formatting a volume but is much faster. It can be used on floppies (even ones with extra tracks), Unidisks, hard disks and some /RAM volumes. The /RAM drivers supported are the Checkmate Multiram driver, the AE Ramworks driver, and the /RAM driver supplied on the Prosel disk.

CHANGE FILE DATE

This routine allows you to change the modification and creation dates on any file - even the volume date (which has a creation date only). You specify the pathname to operate on and you will be shown the existing dates and allowed to modify them. Just hit RETURN to accept the date shown. When you are done you will be given a chance to abort the routine before the new data is written to disk. The main use for this routine is to allow you to make meaningful creation dates for files having no dates or ones on which the date was ruined by use of dumb utilities like the FILER. You do not have to type the dashes or colon shown in the date, any non-numeric character (such as a space) will do, but you must type the data in the correct position on the screen.

CREATE DIRECTORY

This allows you to create new subdirectories without leaving the program. It will even create multiple subdirectories. For example, if you tell it to make a directory called /HARD1/DIR1/SUB3 and DIR1 does not exist on /HARD1, then DIR1 will be created and a subdirectory SUB3 created inside it.

SHOW FILES

This catalogs a directory. Use the arrow keys to scroll forwards and backwards. At the top of the screen is shown the number of blocks used by files in the directory EXCLUSIVE of subdirectories, the number of files in the directory and the number of free blocks on the volume.

SHOW VOLUME NAMES

This looks at all mounted disk devices and shows the slot, drive, volume name, number of free, used, and total blocks, and the creation date of the volume. (Many creation dates will be null. You can rectify that, if you wish, with the "change file date" option.)

TOGGLE BELL

If this is OFF then the bell that is heard at some prompts will be defeated.

TOGGLE PROMPTING

This toggles the prompting state for the FILE COPY and VERIFY. The current state is shown on the menu. If prompts are OFF then deleting of locked files and copying over existing and locked files is done without asking whether you are sure. In addition, copying and verifying files inside selected directories is done automatically when prompting is OFF.

FORMAT A DISK

If the file CD.EXT is in the PREFIX directory when Cat.Doctor is run then the WIPE command is replaced by a MENU CHANGE command and selecting that will bring up another command menu, containing WIPE, and a volume FORMAT command. However, in this case, the small /RAM volume will be disabled.

COMPARE DIRECTORIES

This is also in the CD.EXT file. It will compare the content of two directories (not the contents of the files in the directories). A file listed on only one of the two columns is in that directory but not in the other. Files in both directories but of different file types are shown with their types, and those with different modification dates are shown with their dates.

MOVE FILES

This is also in the CD.EXT file. It will move files (put them on the destination and remove them from the source) on any given volume. This is very fast and can move whole directories. It does not write new files, but just moves the pointers to the files to the new directory.

CONTROL RESET

A control-reset will send you back to the menu. Never do this when any writing to the disk is happening or you could lose substantial data from your disk. (This applies to any program.) If you press control-reset, there will be a brief read from the disk. This is normal - the program is just gathering some information it needs.

<<< Using a mouse with CAT.DOCTOR >>>

A mouse can be used to select options and files. It works a little differently than most mouse interfaces since it "wraps" around the screen. (Thus, for example, if the cursor is at the top of the screen then an upward mouse movement will move the cursor to the bottom, as does an up arrow.)

You can use the mouse button to select options from the main menu or to select or de-select files when a list of files shows on the screen.

When asked for a pathname then the mouse button is interpreted as a "?" (when that is accepted). The mouse can be used to move the cursor across a pathname. Thus you could, for example, use the mouse to click on "Catalog" on the main menu, then click again when asked for a pathname. This will bring up a list of all volumes on line. Clicking on one will produce the directory tree, and positioning the cursor and clicking on the tree will produce the desired catalog of that directory. Move the mouse up and down to scroll the catalog if it is large enough. A final click, anywhere, returns you to the main menu.

In the tree structured directory display, the mouse button is interpreted as a RTN to select the currently displayed directory, and mouse movement is accepted in place of cursor keys.

In a sort, the mouse button can be used in place of the open Apple key to move, with the mouse, files around the screen.

In a catalog, you can use the mouse to scroll the catalog (as the up/down arrows do). The mouse button returns to the main menu.

<<< User modifications to CAT.DOCTOR >>>

Starting at the fourth byte of the CAT.DOCTOR file there are some configurable flags and data. The fourth byte is a $02 and is the a time delay for the name blinking that occurs during prompting. The next two bytes are $00 and are the bell flag default and the prompt flag default. To change either of these, use only $87. The next byte $01 is the printer slot. This is followed by the number of characters per line and the number of lines per page for the printer in the type function. Next is a flag (currently $E5 which, if changed to $65, will cause a catalog line to be printed as a header of typed files. The 5 nibble is the left margin for TYPE (defeated by holding down open apple) and the $20 bit defeats word wrap if cleared (or hold down closed apple at the printer question). The next byte is $60 which is the device number of the default for the format function.

After this is a list of file types in negative ascii followed by the type in hex. This is the same format as discussed for INFO.DESK. As there, the list ends with a number of ascii "NON"s followed by zero bytes. These can be replaced by file type names and the types, and the program will automatically use these when displaying these types.

GETTING RID OF PROBLEM FILES

Sometimes some glitch in the system will ruin a file to the extent that it cannot be deleted by ordinary means. There is a "secret" (meaning not shown on the screen) provision in CAT.DOCTOR that will allow you to get rid of such files. It is in the SORT routine, even though it has little to do with sorting. If you highlight the LAST file in the list and press Open-Apple DELETE then that file will be deleted from the list. Nothing happens on disk until you press return and ask for the "sorted" directory to be written to disk. To delete a file which is not the last one on the list you must move that file to the end, using the arrow keys with Open-Apple, and then press Open-Apple-DELETE.

This only gets rid of the file as far as the directory is concerned and does not free the blocks used by the file and release them to the system. To do that you should then use that provision in MR.FIXIT.

Please note that files deleted in this fashion are not recoverable. You must use it with extreme caution. Entire directories can be deleted this way, by using it on the name of the directory.

For technical reasons this method does not work on a file that is the only file in its directory. However, the directory containing it can be deleted.

BLOCK WARDEN

The BLOCK.WARDEN program is a block editing facility. It requires the Apple 80-column card and a //c or enhanced //e. A similar version BLOCK.WARDEN.UT requiring an Ultraterm card in slot 3 and using an 80x48 display is on the /EXTRAS disk.

When BLOCK WARDEN is run it will read and display block 2 of the device it came from. The program is entered in "R/W mode". The Ultraterm version displays the entire block at once. The regular version displays only one memory page at once and you must use the "\" key to flip between the two pages of a block in both R/W mode and EDIT mode. (In EDIT mode however, moving the cursor will automatically adjust the page displayed.)

R/W MODE

In this mode of BLOCK WARDEN the arrow keys read the next or previous block. The down arrow and up arrow are equivalent to the left and right arrows respectively in the Ultraterm version but they read next or previous PAGES in the Apple screen one.

The Q key will ask if you want to quit.

The R command allows you to specify the next block to be read. Block input is in hex unless you backspace over the $ sign, at which point you can enter a decimal number.

The W command will allow changing the block. (Just hit RETURN for no change.) Then it will ask if you really want to write block as an extra safety to prevent accidents.

The C command allows you to change the slot and drive parameters. Note that this command will not read from the disk, so that you can transfer a block from one disk to another by use of it. Subsequent reads, however, will come from the new disk.

The P command allows you to specify the prefix. This is mostly for use with the F and I commands, but it can also be used for changing devices. (Note, however, that the new disk is read at this command, so that this cannot be used in copying blocks between disks.)

The F command asks for a pathname (full or partial) to be followed. After issuing this command the file name being followed will show at the top of the screen and the arrow keys will move between blocks OF THE FILE. Hitting the ESCAPE key (and some other commands) will cancel the follow mode.
This facility supports all file types including directory files and sparse files. It reads only the data blocks of a file, so it cannot be used to look at the file's index blocks. While following a file you may go to edit mode, edit, return to R/W mode, write a block, and continue to follow using the arrows.

The I command asks for a file name and then shows the file parameters which appear in the directory entry of the file in a form that is easy to read. The bytes on the left give the offset location of the data in the block buffer. When you hit a key the program reverts to the block in effect when the command was issued. If the file is a subdirectory, however, the header of the subdirectory is shown after the first key press. You must note the block number containing this information if you are going to want to use the editor to change any of this information.

The L command disassembles the current buffer contents. [In follow mode with a TXT file it lists in ascii instead. This can be forced in follow mode for any file type with the " command.] The ascii equivalents of bytes appear after the disassembly line. A total of 40 lines (80 in the Ultraterm version) are listed on each page. You are asked for a starting byte (0-1FF) to start the disassembly on. If you just hit RTN this defaults to zero. If you hit the right, up arrow or RTN then the next page of the disassembly is shown; any other key cancels this mode. If you are following a file then the right or up keys pass to the next block of the file when the buffer listing is done and that block is listed. While in the List mode (Apple screen version only) you can dump the screen to a printer by typing Open-Apple-D. This assumes the printer is on and in slot 1. (If a previous Dump command was issued and you selected a different slot for the dump then that slot will be used instead.) The disassembly supports the whole 65816 opcode set. The "M and X flags" attempt to follow the program logic, as in the Merlin-pro assembler. At the start of each page these flags can be reset by using the RTN key instead of the arrow keys and pressing, at the same time, the open Apple key to set M=0 and/or the closed Apple key to set X=0.

The D command dumps the buffer contents to a printer which is assumed to be in the slot of the number key you press when you are prompted to turn the printer. You can abort the command by hitting the ESCAPE key.

Finally, the ^ command sends a list to the printer of all possible "index blocks" on the disk (starting with block 7). This is intended as an aid in a last ditch attempt to repair a blown directory. It would be better if you have an up to date list made by the INFO.DESK program. Also note that if block 2 is bad (so that the volume name is shown as "?") then this command will not work - it will print ALL the blocks on the disk. Thus you must repair block 2 first to the extent that it has a valid volume name and the correct number of blocks (bytes $29,2A of block 2). The list printing can be aborted by hitting the ESC key. The list will include some blocks that are not really index blocks, mostly partially full blocks at the ends of files.

EDIT MODE

In edit mode the arrow keys move the cursor (shown in inverse). Any key other than a control character will be regarded as a change to the editing buffer. (In hex mode it is disregarded if not a valid hex digit.)

The ESC key returns to R/W mode. The ^X key cancels any changes you may have made on the current block. (This is done be rereading the block.) The TAB key toggles between hex and ascii editing modes. You can tell what the current mode is by seeing where the cursor is placed. In ascii editing mode, the high bit of a typed character will be off unless you press the Open Apple key at the same time, in which case the high bit will be on. Control characters can be inserted into the buffer only in hex editing mode.

The ^F key is a lead-in to character find mode. If the cursor is in the hex portion of the screen then the cursor will disappear and you are expected to type two hex digits. (An invalid digit causes this mode to be canceled.) The resulting byte then becomes the "find character" and the next occurrence of it in the buffer will be found and the cursor moved there. (If none exists the cursor returns to the first byte and the "Find mode" message is erased.) After this first find (that is, when "Find mode" is on the screen) subsequent ^F commands find further instances of the find character in the buffer. This mode can be cancelled with the RTN key.
If the ^F key is issued when the cursor is in the ascii portion of the display then an ascii find character will be gathered. (High bit of the find character will be off unless Open Apple is pressed when the character is typed.)

The ^S key selects global search mode. When used the first time this brings up a request to input a search string. If the string you give begins with "$" then it will be taken as a hex string for the search. Otherwise the string is taken as an ascii string.
If FOLLOW mode is in effect then only the followed file will be searched for the string, otherwise the entire disk, from the present position, will be searched. The ascii search is not sensitive to the high bit of characters, nor is it case sensitive. (Thus, for example, a search string of Junk will find both "JUNK" and "junk".) You can cancel a search in progress with the ESC key while the disk is reading, or during input of the search string. (On the Ultraterm version, use ^C for the latter.)
Strings that cross block boundaries will be found by this facility. A melodious beep is heard when a string is found.
After the first instance of the search string is found, others will be searched for upon pressing the ^S key in edit mode.
Any block read with the R/W mode Read command will cancel the search mode. You can also cancel it by hitting RTN in edit mode. The search mode with the same string can be reinstated by typing another ^S command in edit mode and just hitting RTN when the default string is shown.

If the Volume name at the top of the screen, is shown to be "/?" this means that the program encountered an error in trying to determine the name. This almost certainly means that there is something wrong with block 2 (the first block shown). It could also mean, however, that the disk is not a ProDOS disk.

CHANGING STARTUP NAMES

BLOCK WARDEN (Apple screen version only) has a built in feature to simplify entering STARTUP names in SYS files. To change the default STARTUP file name in a SYS file, follow these steps:

1. In R/W mode, type F to follow a file, and specify the appropriate file name. (You may want to use the P command first to set the prefix.)

2. When the first block of the file is read, type the E command to go to edit mode.

3. Use the right arrow key six times to move the cursor to the startup position. At this point a box will be displayed asking for the startup name. (If not, then either the file is not of SYS type, you are not in follow mode, or the file does not support startups.)

4. Type the startup name ending with RETURN. When done you will see the name you entered has been placed in the edit buffer with the appropriate length byte supplied automatically.

5. Type ESC to go to R/W mode, then the W write command to write the buffer to disk, etc.

SUMMARY of BLOCK WARDEN commands:

R/W mode commands:

```
Q....... quit program
E....... go to edit mode
Arrows.. read next or previous block or page
R....... read block (input)
W....... write block (input)
I....... get file info (input)
",L..... list/disassemble buffer contents (input)
          (continue list with right/up arrow or RTN)
          (RTN with Apple keys operate MX flags)
P....... set prefix (input)
F....... set "follow" file name (input)
C....... change device (input slot & drive)
D....... dump block contents to printer
^....... send index block list to printer
```

Edit mode commands:

```
ESC..... return to R/W mode
Arrows.. move cursor
^X...... cancel changes to buffer
^F...... find chr next typed in current buffer
          (or continue find)
^S...... global disk or file search for string
          (or continue search)
RTN..... turn off find and search modes
(Non-control keys are taken as buffer edits.)
(Open Apple key after ^F in ascii mode or during buffer
edits sets the high bit of the character typed.)
```

BEACH COMBER

CAUTION: This program is extremely dangerous. If it misfires in any way or if you have a glitch in your system such as a slight ram problem or if you have a power outage while it is operating it can destroy the entire volume it is working on. You absolutely MUST have a full backup before using it. Destruction may also occur if the disk contains bad blocks or damaged data in a directory. Don't blame me for the faults created by other utilities such as one of the multitude of bad directory sorters.

I TAKE NO RESPONSIBILITY OF ANY KIND CONCERNING THE PROPER WORKING OF THIS PROGRAM. YOU ARE ON YOUR OWN.

---> Make a BACKUP first !!! <---

BEACH COMBER is a ProDOS volume optimizer (particularly for hard disk). It will put all directories at the beginning of the disk, and all files will have contiguous data blocks. This makes for much more efficient file access.

The program will ask for a slot and drive. It will then read and show the volume name in that device and ask if it is all right to continue.

The program will take quite some time to operate. If you absolutely must abort the program, you can use the escape key. The program will take a few moments to react to this key because it will only quit at a time when the disk is "clean", and must also do some last moment writing to disk at this point. If you later want to continue, the program will quickly arrive at the spot where it stopped, for the simple reason that it has little to do before that spot. DO NOT interrupt the program with RESET; this will probably make much of the volume unusable, even though the program does its best to recover.

The program makes 4 "passes" numbered 0 to 3. These are passes through the directory, not the disk. The brief pass number 0 just reads data from the disk, and an error at this phase will not affect disk contents. Pass number 1 relocates the directory and is not very long. However, the program logic in this section is extremely complicated and any obscure bugs are likely to show up at this time. [After all, directories have not only forwards and backwards pointers, but also upwards and downwards ones. Moving one directory block takes a lot of housekeeping.] Pass number 2 relocates all files except for tree files. This pass takes the great majority of the time used by the program. From this point, the screen will show the current block being processed. Finally, pass number 3 relocates tree files, if any. All tree files will be relocated following all standard files. (This refers to the files themselves, not to their directory entries, which will not be changed.)

The program has some limitations. It is limited to volumes that have no used blocks above 10MB. It is also limited to 255 tree files. Since that many tree files, actually filled with data, takes up over 32MB, this is not much of a limitation, but it could be exceeded by a lot of "sparse" files. The program counts trees on pass 0 and will exit without harm if this limit exceeded.

The program needs a lot of ram space and so it will run only on 128k machines. The program does not need, or use, any free space on the disk.

Although I have gone to great lengths to make the program as fast as I can, it will still take a long time to execute on any sizable hard disk. About 5 to 12 minutes per megabyte is typical for the first run. Later runs may take much less, depending on what areas of the disk have been changed and how extensive are the changes.

I don't think you should use this program too often, although that is a matter of preference. Once every two or three months seems about right to me.

The program goes through directories and files in the same order as the INFO.DESK program. Thus, to speed later uses of this program, you can put directories containing files that can be expected to change, late in this order. (I.e., late in the main directory.) Put stable directories early. Of course, this may not be practical for all people and it may not speed things up all that much anyway.

The disk also contains a 20MB version BEACH.COMBER.20 that requires a large ram volume to work. It requires a ram volume with more than 512 contiguous blocks free. It will work with the GS /RAM5, with the ProSel ram driver and the AE Prodrive, but not with the Checkmate driver which has "illegal blocks".

There is also a full 32MB version BEACH.COMBER.32. This one runs ONLY on a GS with more than 512K free ram (checked by the program). It does NOT use /RAM5 so it is best to have a small RAM5 or none at all.

If BEACH.COMBER encounters a problem it will shut down and display an error message. Some error messages like Block Read error are self-explanatory. Others like Block Reference error are rather technical and just generally indicate some defect in the volume (or may be caused by a hardware problem). One error that seems to occur more than others is a "Two owners" error. This error occurs only on the initial pass before any writing is done to the disk. It indicates that there are two files, or a file and a directory or bit map block, that contain the same block. Thus one or both of the files has been corrupted. This is a serious error and this condition should be dealt with as soon as possible and as carefully as possible. Do not write to the disk until it is cured. Mr.Fixit should report a "Block in use" error (meaning the same thing) and "File skipped". That will indicate one of the two bad files, and Info.Desk can be used to determine the other file. Such files MUST NOT BE deleted by ordinary means. Use the "problem file deletion" method described at the end of the documentation for Cat.Doctor.

MISTER.FIXIT

This program (file MR.FIXIT) is a directory repair utility. There are four modes: a Test mode in which nothing is altered on the disk, a Fix mode which attempts to correct defects found in the directory structure, etc, a Main directory mode which attempts to reconstruct the main directory and a Bad blocks mode which scans for bad blocks and, if desired, places them in a bad block file. The Fix and Main directory modes are very dangerous and can change a bad situation to a worse one, so never use them without first using the test mode and never use them without an adequate backup that you can fall back on.

The program tests and, if fix mode is active, fixes the following defects on any ProDOS volume:

1. Header pointers of all active files.
2. Parent pointers and parent entry numbers of all subdirectories.
3. Backwards directory links.
4. Used blocks marked free in the bit map.
5. Illegal characters in file names.
6. Entry length (the program assumes this should be $27).
7. Number of entries per block (assumes this should be 13).
8. File count in each directory.
9. Incomplete deletes (deleted files with non-zero "name length").
10. Incorrect directory storage types.
11. File and directory block counts.
12. File and directory dates and times.

In file names (point 5) lower case characters are converted to upper case, high bits are stripped, and other illegal characters are replaced by periods. Illegal dates and times are zeroed.

The following items are checked and reported, but not acted upon:

13. Blocks used by two or more files.
14. Block number out of range (past volume size).
15. Unknown storage types.

Some errors result in files or directories being skipped over. You will be told if this occurs. It happens because of information damaged in such a way that the situation cannot be handled, or the damage is such that the supposition is that the remaining data is invalid.

Some block read/write errors cannot be handled and will result in termination of the program.

Sometimes in the error printout you may see a directory name printed twice. This is not a bug; it indicates that the correction refers to the directory "header" rather than the "parent block".

MR.FIXIT will also look for blocks that are marked used on the volume bitmap but are not used by any file. You will be given the option, in Fix mode, of releasing these blocks. Sometimes areas of a volume are marked off without belonging to any file and in this case you should not ask that these blocks be freed. Examples are the UNODOS area on a Unidisk or a Pascal area. As a general rule, if the number of blocks indicated as being marked, but unused, is large, then you should assume that they are marked for a reason and should not free them. (UNODOS, for example, marks exactly 400 blocks.)

The most important block on a ProDOS volume is block 2. Since it is accessed much more often than any other block, it is also the most likely to be damaged. If it is, you will probably see a message in MR.FIXIT that block 2 is too damaged for the program to function. There is a special provision for attempting a repair of the main directory, and it is accessed automatically when you request Fix mode and block 2 has extensive damage. You can also force this mode by selecting M at the Test/Fix prompt. You will be asked if the program should assume that the "bit map" is valid (default = Yes). Ordinarily you should select Y unless there is some reason to believe the bit map is damaged. (For example, if you are trying to resurrect the subdirectories after a disk has been "wiped" by Cat.Doctor then the bit map will NOT be valid and you must select N at this prompt.)

When this mode has been selected by M or automatically because block 2 has been determined to be substantially damaged MR.FIXIT will tell you that this attempt is being made, and will give a few particulars along the way. Although the routine can resurrect most subdirectory pointers, it cannot do anything for standard (non-subdirectory) files in the main directory. While this routine is operating, every block on the volume will be read. This may take some time, so be patient. When it is finished, you will get the "Another?" message. Note that this routine does not do the other repair jobs mentioned above. You can run through the disk test again to check if those things are all right. You should realize, however, that this main directory repair routine expects most of the rest of the volume to be reasonably valid and normal. (For example, it assumes that the "bit map" starts on block 6 if the main directory header has been damaged. This is true for almost all disks, but not for some RAM volumes; this routine should not be attempted on a RAM volume or any other volume you suspect is organized in an unusual way.

If you select the Bad block mode then you will be asked if you want only to test or to fix. If you select test then the volume will simply be scanned for bad blocks and the results reported. If you select fix then there will be an attempt to place the bad blocks found in a bad blocks file. This file will be created in the main volume directory (so be sure there is room for it). If a bad block is a data block and not a directory or index block then several attempts will be made to read it and relocate it. If it cannot be read then a fake block is substituted and marked with the message "DAMAGED BLOCK".

MISTER FIXIT can be used to resurrect the subdirectories in the main directory and all other files that are not in the main directory. (It is not possible to retrieve the vital pointers for other files in the main directory, although those files may still be intact on the disk somewhere.) This can be used on a disk that has been inadvertently "wiped" by Cat.Doctor or "deleted" by COPY][+ or a large volume that has been "formatted" by filer (i.e., as long as the disk was not physically formatted). To do this, first use the option M of MR.FIXIT to repair the main directory. Then use the T mode to check the types of remaining errors. (There should be a LOT of block free errors, and a file count error.) Then use the F mode to fix the remaining errors if there are no fatal errors. You must remember that this program makes decisions that sometimes may be inappropriate and may leave some undetected problems.

MISTER FIXIT can be automated by putting the desired key sequence after a semicolon in the startup position of the MR.FIXIT file.

NOTICE

ProSel-16, ProSel-8 and their support programs are not "shareware".

The source listings of these programs run to over 60,000 lines of code and fill binders over six inches thick. If you want to support the great effort that goes into creation of such a software package, you will respect my right to a fair return on such a major project. It is fine to show the programs to others, but to give them away is illegal and immoral.

Updates to ProSel-16 for current owners of ProSel-8 are $55 first time.

The price of ProSel-8 alone is currently $40 and updates are $10.

These prices are subject to change without notice.

Glen E. Bredon
521 State Road
Princeton, NJ 08540
(609) 924-5976

I am often away during the summer months (June 1 - Sept. 1) and processing of orders may experience considerable delay at that time.

PRODOS, GS.OS, START.GS.OS, PRO.FST, TOOLSETUP, TS2, APPLEDISK3.5, CONSOLE.DRIVER, P8, ERROR.MSG and BASIC.SYSTEM are copyrighted programs of Apple Computer, Inc., licenced to Glen Bredon to distribute for use only in combination with ProSel. Apple Software shall not be copied onto another diskette (except for archive purposes) or into memory unless as part of the execution of ProSel. When ProSel has completed execution Apple Software shall not be used by any other program. Apple is a registered trademark of Apple Computer.

APPLE COMPUTER, INC. MAKES NO WARRANTIES, EITHER EXPRESS OR IMPLIED, REGARDING THE ENCLOSED COMPUTER SOFTWARE PACKAGE, ITS MERCHANTABILITY OR ITS FITNESS FOR ANY PARTICULAR PURPOSE. THE EXCLUSION OF IMPLIED WARRANTIES IS NOT PERMITTED BY SOME STATES. THE ABOVE EXCLUSION MAY NOT APPLY TO YOU. THIS WARRANTY PROVIDES YOU WITH SPECIFIC LEGAL RIGHTS. THERE MAY BE OTHER RIGHTS THAT YOU MAY HAVE WHICH VARY FROM STATE TO STATE.