

COMMERCIAL-IN CONFIDENCE

PERQ Hard Disk Primer

This document gives a short introduction to the PERQ hard disk and the format used by P.O.S.

The disks used on PERQ are manufactured by Shuggart. They come in two versions: 12 and 24 megabytes. Each disk is divided into sectors. Each sector contains is divided into three sections: a) physical header, b) logical header and c) data.

The physical header contains:

Sector
Track
Head

The logical header contains information that is useful to the file system. The information kept in the logical header is:

File Serial Number	2 words
Logical Block Number	1 word
Pointer to previous block	2 words
Pointer to next block	2 words

The data portion of a sector contains 256 data words.

POS File System Format

P.O.S provides a simple file system. The first 188 sectors of the disk are used to hold the boot file. This is the disk file that is read by the boot micro-code at boot time. Following the boot file are 180 (360 if the disk is 24 megabytes) files. Each file contains 129 sectors or blocks. Each file has a file ID associated with it. The file ID of the boot file is -1. The file ID of the user files are the integers 1 through 180. File ID 0 is reserved to mean no file. The blocks of each file are numbered -1 through 127.

Block -1 of each file (except for the boot file) contains some directory information for that file. The first word of block -1 is the number of blocks that are in use in the file. This number does not include block -1. The second word in the block is the number of bits that are in use in the last block of the file. Starting at word three in block -1 of a file is the name of the file. This is represented as a PERQ string. The low order byte of this word contains the length of the file name in characters. Following this is the name of the file.

The first file, file ID 1, is reserved for use by the file system. This file contains the directory for the disk. This directory is only used as a hint. The information in this file is checked against the information kept in block -1 of the file.

What can happen to you disk

There are a number of things that can happen to your disk. They will be listed here and described in detail latter.

- 1) Information in the data portion of a disk sector is destroyed.
 - a) File data is lost.
 - b) The directory informaion in block -1 of the file is corrupted.
 - c) The directory information in File ID 1 is destroyed.
 - d) The boot area of the disk is destroyed.
- 2) Information in the logical header of a sector is destroyed.
- 3) Information in the physical header of a sector is destroyed.

How to fix your disk

This section gives some procedures for putting your disk back together. If you do not understand any part of these procedures you should seek help. It is possible to blow away your disk if you are not carefull. If at anytime during the execution of any of these procedures you get disk errors that are not recoverable you may have to go to another of the procedures and then restart the procedure that you were working on. The destruction on information in the logical or physical header of a disk sector will keep you from accessing that sector. Physical header errors usally generate I/O errors -7 and -8. Logical header errors usally generate I/O errors -9, -15 and -16.

There are a number of places in these procedures that require you to run a program. If the disk is in a state that you can not run a program then you must boot a stand alone program into the Perq from the floppy. There are three boot floppies that are provided with each Perq. They are:

POS Boot	(Boot the operating system)
DiskProgBoot	(Boot the program to fix disk errors)
PLXBoot	(Boot the program to read floppies)

To boot one of these programs place the floppy into the drive and press the boot button. Booting from floppy is not very fast. You should go get a cup of coffee if you need to do this. DiskProgBoot provides the functions needed to repair a broken disk. It has the facilities of the File Utility, DiskFix and DTst.

Data in the file is destroyed:

There is no way to recover from the loss of file data. Keeping a backup of the file is the only way to assure that this is not a problem. It is very rare that file data is lost.

Data in block -1 of a file is destroyed:

There are two common ways in which the data in the directory block of a file is corrupted.

The first is that the words that give the number of blocks in use and the number of bits in the last block are wrong. This is usally caused by exiting the editor with a "Quit, Update" and the typing two <ctrl>C before it is done. To fix this problem you must correct these two words. The procedure for doing this is:

1) Find the File ID of the file. This is done by doing a directory of the file in the File Utility.

2) The next step is to use the Block Write function of the File Utility to change words zero and one of block -1 of the file. To do this type:

```
BlkW <File ID> -1 0 <cr>
```

3) Now fill in the first two words of the block. This is done by typing:

```
128 <cr> (This is the number of blocks)
```

4096 <cr> (The number of bits in the last block)
<cr>

This has now set the size of the file to be 128 blocks and the last block of the file to be full.

4) Now edit the file and remove the junk that you will find at the end of it.

The second type of error that occurs in the information in block -1 of a file is that the name of the file is destroyed in some manner. In order to fix this problem do the following:

1) Find the File ID of the bad file. This is done by entering the File Utility and doing a directory. The File Utility will generate an error message at the file that has the bad directory data. The File ID of the file that generated the error is the ID of the file that you want to fix.

2) Change the name of the file. This is done using the block write command in the File Utility. Type the following command to the File Utility:

BlkW <File ID> -1 2 <cr>

3) You are now ready to write the first word of the file name. It is suggested that you pick a short name for the file. The simplest one is "A". To write the name "A" into this file you must create a Perq string. This is formed by putting a byte count in the low order byte and the first character of the string in the high order byte. To make the file have the name "A" you must put a 1 in the low order byte and an octal 101 in the high order byte. The decimal representation of this is 16641. To place this number into the third word of the directory block type:

16641 <cr>
<cr>

4) You have now changed the name of the file to "A". Type:
Scavage
to the File Utility to update the directory file. At this point you can go back and figure out what the file name should be and use the Rename command to change it.

The data in the directory file, File ID 1, is destroyed:

To fix this problem enter the File Utility and execute the Scavage command. If during the execution of the Scavage command you get an error of the form "String to be assigned is too long" then there is an error in the directory information of some file. See the section on fixing Block -1 errors and then repeat this procedure.

The boot file is destroyed:

If your machine is running POS all that you have to do to fix the boot file is run the WriteBoot program. See the documentation on WriteBoot for more information on how to do this. If you try to

boot your machine and the DDS indicated that there was a boot file error then you must boot POS from floppy. Simply place the floppy that contains the POS boot file into the drive and hit the boot button. After the machine boots you can run WriteBoot and rewrite the boot file. If you get disk errors during the execution of WriteBoot then the logical or physical header information of some of the boot file sectors is destroyed. See the sections on fixing logical and physical headers for details on how to fix these problems. After you have fixed the problems restart this procedure.

The logical header information for a sector is destroyed:

The logical header information for a file is recreated with the DiskUp program. To fix the logical header information of a sector you must first know the File ID and block number of the bad sector. If you already know the File ID of the bad file the Check command in DiskUp will tell you what block is bad. If you do not know the File ID then the DiskUp Search command can be used to find the bad file and sector. Once the bad sector is found it can then be fixed. The first thing to try is the DiskUp Fix command. Type:

```
Fix <File ID> <Block>
```

DiskUp will then attempt to fix the bad sector. The Fix command first tries to read the bad sector. If it can not read the sector it will ask if it should go on and try to do a diagnostic read. If this read fails then you are asked if it should try to do the write. Once the command completes you can check the results with the Check command. If the block was not fixed then you can try a standard DiskUp Write. If this fails then there is an error in the physical header of the disk block. See the section on fixing physical header errors for more details.

The physical header information for a sector is destroyed:

The program DiskUp is used to fix the physical header information on the hard disk. It is also used to bring up the disks when they come from Shuggart. DiskUp is loaded into a machine from the DiskUp boot floppy. To run DiskUp place the DiskUp Boot Floppy into the floppy drive and hit the boot button. At this time DiskUp will start running.

The Search and Check commands can be used to find the cylinder that has the bad header. If you know which file it is then use the Check command. If you do not know the file the Search command will read the entire disk looking for bad cylinders. The error information that is produced when an error occurs gives the File ID, Head and Cylinder where the error was. To fix the error execute the Format command using the Head and Cylinder that was obtained from the Check command. Format will format the portion of the disk that is defined by Head, Cylinder. After the cylinder is formatted you will have to execute the Init command on the file that resides on the cylinder that you just formatted. It is possible that more than one file was effected by formatting a track on the disk. To test for this you should execute the Check

command on the files that reside around the file that you just initialized. If either of these files are bad then simply use the Init command on them.