Summary of Stacking FileStore (SFS) Dealer Test Software

Introduction

This document describes the contents of the Stacking FileStore dealer test software disc. It is intended for use by engineers troubleshooting and correcting faults found in the Acorn Stacking FileStore, units E01S, E40S and E60S.

The software is provided on a 5.26in disc containing the following files:

!Boot	Start-up file
ReadMe	This file
FServMenu	Main menu, used to run the other programs
FServCMOS	CMOS RAM editor
FServCopy	Sector-by-sector disc duplicator
FServEdit	Disc structure editor
FServFmt	Formatter for FileStore Winchester discs
FServInit	Initialiser for FileStore Winchester discs
FServSoak	Disc and Econet soak tester
FServTest	E01S circuit board tester
FSDB	Machine code for use by FServSoak
P1	Parameter file for use by FServSoak
Library	Library containing system utilities
Source	Directory containing "uncrunched" source

Hardware required

The Stacking FileStore dealer test software is for use on a B B C Master 128 Microcomputer with ADFS and a 5.25in floppy disc drive, on a network connected to the FileStore to be tested. (The network connection may be via a Bridge) An Econet Test Box is also needed for FServSoak and FServTest. To use this, make the following connections:

From: B B C Micro, "User port"
To: Test Box, central (unlabelled) socket
Using: 20 way parallel connector cable

From: Test Box, "Tester" socket
To: the Econet containing the B B C Micro
Using: standard Econet cable

From: Test box, "UUT" socket
To: the Econet containing the FileStore
Using: more standard Econet cable

This should not be done on a network that is in active use - one of the purposes of the test box is to simulate a poor Econet connection by transmitting noise on the line. This would annoy other users. If it is necessary to modify the programs for any reason, it is most easily done with a Second Processor, connected to the B B C Micro Tube. This should be attempted only as a last resort, and after Acorn have been contacted. The programs as provided are in a highly compressed ("crunched") form, which is not particularly easy to change (although in the absence of a second processor, there is no choice). The directory $.source contains the programs as they were before they were crunched, along with a file called DoCrunch that generates the compressed versions. (Type `*exec docrunch' this uses the programs HiBASIC and Crunch from the directory $.Library.)

!Boot

The main menu of the Stacking FileStore dealer test software is invoked by putting the disc provided into the disc drive of the B B C Micro and pressing Shift-Break, or else by typing *exec !boot to the BASIC prompt. These have the same effect: the file !boot is executed, and chains the main program of the suite, FServMenu.

FServMenu
This will first ask you which FileStore to use: type in a station number - either the number of a station on the local network, or else a net number followed by a full stop and a station number on that network (if you are using a network bridge). You will be logged onto that station, if possible, and be offered a menu containing the remaining programs of the suite; otherwise, an error message is printed, and a truncated menu of just "Help," "Logon," "Quit" and "Star commands" will be offered, so that you can take whatever corrective action is needed.

Common features of all programs

Menus
All programs in the suite use the FileStore chosen when FServMenu starts up; this may be changed by the "Logon to new station" ("L") option in FServMenu. Further, they share the idea of the "current drive," which may be changed by the "Select drive" ("D") option (in those programs for which it is relevant), and examined by the "Drive information" ("I") option (ditto).
All programs also have a "Quit" ("Q") option, which takes you back to the main menu (or from the main menu back to the BASIC prompt) and a "Help" ("H") option, which prints the option list.

Input
Menu entries are selected, and yes/no questions are answered, by a single key-press: an invalid choice has no effect. When you choose a menu option, Space and Return are both equivalent to "Help" ("H").
Numerical input may be given in decimal or hexadecimal (preceded by a "&"): some questions supply a "&" after the prompt automatically: this may be deleted if you wish to give the number in decimal.

FServCMOS
This is a non-volatile (CMOS) RAM editor. It reads a CMOS RAM address (in the range &E to &3F) and then prompts for a new value, which it writes at that address. It possible to leave the CMOS RAM in an inconsistent state, as some of the entries are check-summed. You are warned if this happens.
Many of the fields of the CMOS RAM have FileStore commands to alter their contents: these are the station number, MaxUsers, MaxDrives, the printer port name, PrPage, Copyright check and FSUser.

FServCopy
This program copies the entire contents of a disc from one drive to another. The discs must both be the same "shape," i e., the same number of sectors, divided into the same numbers of cylinders, heads and tracks (this is checked by the program), and the target disc must be formatted (but it need not be initialised, as the initialisation data will be copied as well.
The copy is done sector by sector, and sectors of the disc that are not used by any file are not copied, for efficiency. The result is an exact duplicate of the original disc. The title should be changed before it is used in a FileStore, as it will be same as the disc from which the copy was made. This can be done using FServEdit, or the FileStore command *FSNAMEDISC.

To duplicate a disc onto a disc of a different shape, the utility NetMgr (supplied with the FileStore) should be used: this copies the files and directories individually.

FServEdit
A sector-by-sector disc structure editor. This lets you make any kind of modification to the structure of a disc: it cannot be emphasised too strongly that it is completely "insecure," in that it lets you do anything to any sector, with no consistency checking unless you explicitly ask for it. It is very easy to delete data from sectors, or alter directories so that they no longer refer to files (among other mistakes), with this program! Of course, without this flexibility, it would be less useful than it is.

The main features of FileStore discs that it knows about are: directories, map chains, sectors, the cylinder allocation bitmaps and the disc name.

Briefly, a FileStore directory contains, along with the catalogue information about a file, the sector number (called the "System Internal Number" or "S I N") of an object called a `map chain,' which is a list of sector numbers where the data for the file can be found. This also applies to directories. To refer to a file, you refer to the S I N of its map chain.

Each disc has a boot sector (and a redundant copy of this). ServEdit can list the contents of the boot sector: this contains the S I N of the root directory, $ (really, the S I N of the map chain that refers it). Directories can be listed ("List directory," "L") and entries can be added to them ("Add entry," "A") and removed ("Remove entry," "X").

Map chains can be listed ("List map chain," "C").

The contents of sectors can be edited: there is screen editor to allow the user to alter the contents of any disc sector, using the command Edit sector" ("E") or "Read sector" ("R").

The cylinder allocation bitmaps can be listed ("List bitmap blocks," "M") and checked: "Validate disc" ("V") makes a new allocation bitmap (in an ordinary disc file), by following the directory entries on the disc, without reference to the bitmaps on the disc; then "Compare bitmaps" ("K") can be used to compare the bitmaps on the disc with the contents of this file, and "Revalidate bitmaps" ("Y") to replace the bitmaps on the disc with the ones from the file, in the case that they are different.

The name of the disc: "Change disc name" ("Z") is equivalent to the FileStore command FSNAMEDISC.

FServFmt

This is a general formatting program for FileStore Winchester discs (Rodime RO650 and RO3000 series) Disc controller pages may be examined ("Print disc controller page," "P"), and the contents of user pages A, B may be written ("Initialise maintenance pages," "U"): for Acorn drives, these must contain the eight characters `(C)Acorn'. Note that user page B is "write-once:" after it has been written, another attempt is an error. The error recovery parameters of the drive may be changed ("Change error recovery," "R"). See the Rodime User Manuals for full details.

The numbers of heads and cylinders may be specified by the user, if they can't be read from the drive. ("Change drive parameters," "C").

The Rodime self-test facility may be invoked ("Self test," "S"). Again, see the Rodime manual.

The defect list may be examined ("Type the defect list," "T") and changed ("Add defects at logical block," "A").

There is a testing facility ("Sector read/write tests," "4") (with error log, "?") similar to the soak test provided in FServSoak.

The main purpose of the program is to format the disc ("F"). After formatting, it writes an ADFS partition, and will then optionally verify or certify the disc automatically, under the control of "Change format parameters ("M"). The interleave factor may be changed: the default is 8, which is believed to be optimal over a wide range of usage types of FileStore.

Also, the ADFS partition may be written without first formatting the disc ("W"). The ADFS partition is on cylinder 0 and it contains the disc parameters in the form that ADFS understands. This allows the disc to be initialised by FServInit (it is not suitable for use as a FileStore disc at this point).

Verification (checking by reading each sector, "V") and certification (checking by reading and writing each sector, "3") are also available as separate options.

FServInit
This modifies the ADFS partition (on cylinder 0) and the FileStore partition (on cylinder 1) for a disc, based on an existing ADFS partition put in place by FServFmt. After this has been done, the disc is ready for use in a FileStore.

FServSoak
This programme initialises the FileStore with a test program, and then provides various commands to control the execution of that program. The program is in the file FSDB. Because the program can test any combination of discs on the FileStore, it does not use the "current disc" that the other programs use. Instead, it provides a list of "activities," which may be examined with the "View activities" ("A") option. The active activities are executed cyclically, and may be changed with "Modify parameters" ("M"). The current status of the parameters (that is, whether or not they are active)may be saved and loaded on disc ("S," "L"): the initial settings are read from a file called P1, with enables tests `Net GdLne IntCk' (network test with a good line using the FileStore clock) and `Gen Net Comm' (general network communications) only. If any of the disc tests are started, data on those discs will be destroyed.

FServTest
This programme tests many of the features of the E01S main circuit board. These include:

The RAM (4 tests).
The non-volatile (CMOS) RAM.
The Econet connection (2 tests).
The floppy disc controller.
The real-time clock.

Known Bugs
FServSoak sometimes reports an error on reading or writing the sector with a number 1 greater than the largest sector number on a disc. These errors should be ignored. (The number of sectors on the disc can be found by using "Drive Information" ("I") in FServInit, and multiplying together the numbers given under "Heads," "Cylinders" and "Sectors/Track.")

FServSoak sometimes puts erroneous values (of 65535) in the "Errors" and "Cycles" fields of the "View test in progress" ("T") display.

FServSoak gives a spurious "Error &41 on transmit" after it asks you to switch off the E01S.