

**FILE SERVER/PERSONAL FILE**

**Product Specification**

# Contents

Introduction

Overview

Product Configurations

Product Functional Specification

  User-level Functionality

    File Server Mode

    Remote File Server Mode

    Install/Maintain Mode

    Extended File Server Mode

    Personal File Mode

  Interconnect Functionality

    AppleBus

    RS232C

    AppleBus - Extended

    DirectConnect

  Hardware Functionality

Software Structure

Hardware Structure

Development and Product Staging

Projected Unit Cost

Open Issues

# Introduction

This document describes a new Apple product, the File Server/Personal File. The purpose of this document is to give a comprehensive product description across all dimensions.

As such, this document provides a reasonably detailed overview of the product. There will be several other documents covering topics such as detailed hardware design, product introduction plans, etc. This document provides the reader with a complete, technical introduction to the product - it is the "glue" that binds together and interrelates the other more comprehensive specifications. Because of its overview nature, this document will stabilize sooner than the detailed specifications.

References to those detailed specifications will be found throughout this document and can be read as appropriate to the individual reader.

## Overview

As its name suggests, the File Server/Personal File (FS/PF) is a product with two similar but distinct product applications. It can be configured as a node on an AppleBus and shared by a number of users (workstations) - access is at the nominal AppleBus speed of 230.4Kbps. It can also be directly connected to a Macintosh as a dedicated, intelligent, highly-buffered external file system - communication with the Macintosh is ~900Kbps. The basic, online storage capacity is 10Mbytes and can be expanded to 67Mbytes. (A follow on product, the Expansion File, would permit online storage of over 200Mbytes.)

The FS/PF hardware consists of four components: 1) an industry-standard, 5.25" rigid disk drive (10 or 67Mbytes), 2) an Astec power supply, 3) a digital board, and 4) chassis and plastic case. The digital board design is similar to that in the Macintosh with superfluous functionality removed (such as the video support) and functionality added to support the FS/PF such as a DMA rigid disk interface and an interface allowing the addition of tape backup units and additional disk units.

The software which runs in the FS/PF and provides its functionality is based on the Macintosh Operating System. The Macintosh file system is not used since it was not designed to support large capacity disks. - a new file system is being developed. In addition to providing the functionality of a shared file server and a dedicated, external file system for Macintosh, the software allows "applications" to be added for co-execution in the memory of the FS/PF. This will be used to provide mail server or print spooling functionality, for example.

Additional FS/PF file software runs in each Macintosh (or other AppleBus-connected workstation) desiring to use this functionality. For the File Server this software allows the user to store and retrieve files between the Macintosh and the File Server. For the Personal File, there is a small software module that maps file system requests into commands across the ~900Mbyte connection to the Personal File.

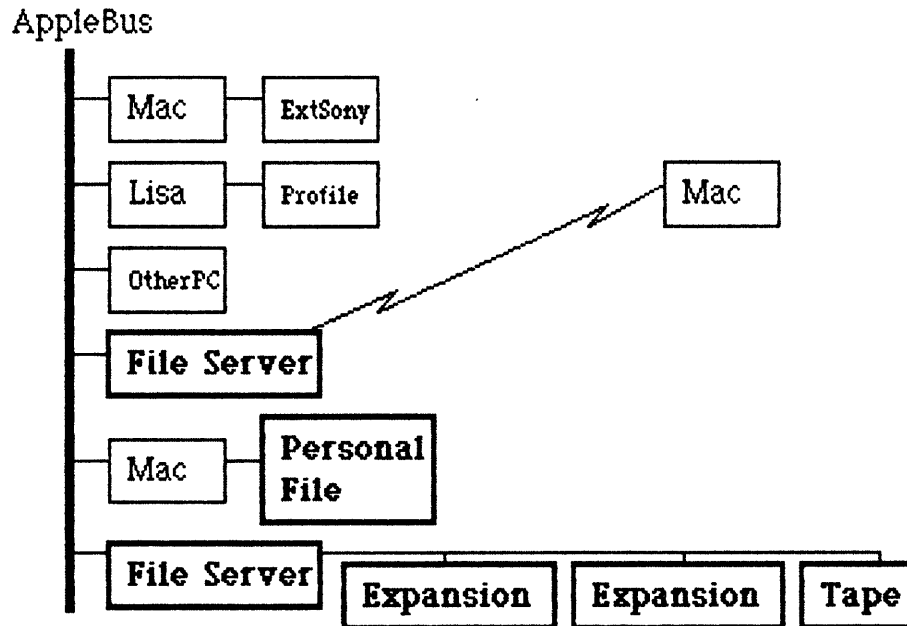
Finally, the SF/PF Install/Maintain software runs on the Macintosh and is used to perform initial installation and configuration. This software is also used to perform routine maintenance such as backup, as well as diagnostic functions.

All software interfaces to the FS/PF will be published. This will allow third parties to develop applications that use the file server in either mode. It will also be possible for third parties to develop applications and have them execute on the

FS/PF.

# Product Configurations

The diagram below shows the two ways a FS/PF can be connected: to the AppleBus as a File Server and directly to a Macintosh as a Personal File store.



Note that these are typical configurations, not every possible one. For example, there are obviously other configurations for a standalong Macintosh beyond those depicted above.

As this diagram suggests the FS/PF is really part of a family of products that includes an Tape Backup unit and Expansion File. The first version of the product will be the single-box File Server. The single-box File Server and Personal File, shown above, will contain a single disk drive - 10 or optionally 67Mbytes.

The Expansion File will be in the same physical box as the FS/PF but will contain a much larger disk, probably about 70Mbytes. The Tape Backup unit will be in a identical box, but with access for inserting/removing tape cartridges. A bus is used to connect these boxes to the File Server (hopefully we will be able to source rigid disks and tapes such that this is a standard bus such as ESDI or SCSI).

Neither the Expansion File nor the Tape Backup are discussed further in this document. They are separate product programs that will follow the FS/PF closely in time and will be described in detail in subsequent documents.

# Product Functional Specification

The functionality of the FS/PF can be described from several viewpoints: that seen by the user at his Macintosh from the product as a whole, that provided by the FS/PF across the AppleBus or direct connection, and that provided by the hardware.

## User-level Functionality - File Server Mode

When connected to an AppleBus as a file server, the FS/PF provides file-at-a-time storage and retrieval of information. In addition, the user may list all or a subset of the files on the file server, rename and delete files. Files that the user wants to operate on with a finer granularity are first transferred to a disk local to the Macintosh. This access will be controlled in password fashion at least at the "file drawer" level (the first level of subdivision of files within the file server).

The exact user interface for these operations is TBD. An overview will be inserted here when more completely defined. The most desirable alternative is via a desk ornament. This would use the "finder-friendly" interfaces to interact with the finder to display files on the server and transfer them between the Macintosh and the file server. This ornament is available in any other context to transfer files between the file server and file volumes directly-connected to the Macintosh.

## User-level Functionality - Remote File Server Mode

The File Server can also be accessed from a standalone, remote Macintosh via standard RS232C communications, either directly connected or via modems and phone lines. The user accesses the file server exactly as if his Macintosh were on an AppleBus in terms of filing operations. The addressing of the File Server (specification of phone number or dialing the phone) will be different.

## User-level Functionality - Install/Maintain Mode

The Install/Maintain program is a standard Macintosh application that is used to control the FS/PF in special, atypical situations. The exact user interface for these operations is TBD. Its description will be inserted here when more completely defined.

A virgin FS/PF is assumed to have an uninitialized disk and no software other than that in its ROM. When first powered up it awaits contact with the Install/Maintain program either over AppleBus or via the direct connection. When this contact is made, the Install/Maintain program downloads a part of itself into the FS/PF. This

is the install/maintain agent.

All subsequent install/maintain operations are performed by the agent as directed by the user through the Install/Maintain program in the Macintosh. These operations include:

- 1: downloading the operational software for the FS/PF and installing it on the rigid disk;
- 2: setting of system parameters - file server name, time, date, etc.;
- 3: addition of Expansion Files;
- 4: performing mirror backup of the disk storage to tape (incremental backup is not planned at this time);
- 5: invoking file/directory scavenging operations after power, system or software failures (may be automatic as well);
- 6: invoking diagnostic programs;
- 7: configuring the Macintosh for access to the FS/PF (either via a file transfer desk ornament or as as an external file system).

For the Personal File typically only operation 1) above will be necessary. Actually software will be installed during manufacturing as part of system test. If this is the same as the operational software, the Personal File may not require any installation beyond physical connection.

### User-level Functionality - Extended File Server Mode

This mode is roughly a combination of the two modes described above. The FS/PF is attached to the AppleBus and can be used by the user as described above under File Server Mode to access entire files at a time. In addition, the File Server provides the functionality that allows software in the Macintosh to open, close, read and write files without transferring them to local disks.

This mode of operation "appends" a hierarchical structure of files to those of the local workstation for use in any fashion. While the appearance to the user, through the Finder or Standard File, will be like that of an additional "volume" of random access information, the FS/PF is not operating as a block server.

Extended File Server Mode could also be described as Remote, Shared Personal File Mode. Note that due to the sharing of the AppleBus, its lower transfer rate and the sharing of the FS/PF that this mode will generally provide much lower performance than the directly-connected Personal File but better performance than that of the microfloppy drive (due to better average access times). This mode will probably be the last to be implemented. It requires additional protocol definition and solving difficult issues of access control and timeout. It also creates a situation where workstations move much of their file I/O "down" the AppleBus - this could overload the bus resulting in poor performance.



## User-level Functionality - Personal File Mode

As a Personal File this product provides dedicated external file storage to a single Macintosh. The Personal File appears to the user as an additional disk drive or volume both in the Finder and in the Standard File window. It is implemented as an external file system "driver" dynamically loaded into the Macintosh Operating System.

Otherwise the distinction between the Personal File and other file volumes (internal or external Sony, for example) is invisible to the user. Due to the computational intelligence, buffering and base speed of the rigid disk in the Personal File, there should be a marked improvement in user-perceived performance when running file I/O intensive applications.

## Interconnect Functionality - AppleBus

At power on, code in the ROM of the FS/PF will dynamically determine whether it is connected to an AppleBus or directly connected to another computer (Macintosh).

In AppleBus mode, the File Server communicates via standard Apple protocols. AppleBus Link Access Protocol (ABLAP) is used for the link layer (ISO OSI level 1), Datagram Delivery Protocol (DDP) is used for the network layer (level 2) and Applebus Transaction Protocol (ATP) is used for the transport level (level 3). In addition, the File Server implements Name Binding Protocol (NBP) which allows user to address it by name rather than by network address. These protocols are described in detail in other Apple documents.

These protocols provide generic reliable transport services. In addition the File Server will implement a File Transfer Protocol (FTP). FTP implements the initiation and termination of the operations provided by the file server, while ATP, DDP and ABLAP actually transfer the data. The capabilities provided by FTP include those to store, retrieve, delete, rename and list files. The parameters to these commands are primarily pathnames with the addition of passwords for access-protected files/directories. The details of this protocol have not been specified, but more details will be added here when it is.

Additional protocol constructs will be defined for downloading of software and running diagnostics on the FS/PF.

Of course, these are the protocols used by the File Server software. Other applications running in the File Server can use their own versions of any of these protocols except ABLAP, in the tradition of the ISO OSI model. However, the Apple protocols will be in the ROM and will probably be preferred when they are suitable.

## Interconnect Functionality - RS232C

The physical interconnect when accessing the File Server from a non-Applebus machine is the RS232C standard. File server operations will be performed using exactly the same protocols mentioned above - FTP, ATP and DDP. A different link access protocol (AsynchLAP) will be developed as appropriate for RS232C. While the functionality of ATP and DDP is probably overkill for this application, we will use it anyway to avoid developing new protocols. ATP and DDP will already exist in both the Macintosh and the File Server.

## Interconnect Functionality - Extended Applebus

When the File Server is providing both basic service (file at a time) and extended service (full file system access) it supports the protocols described above (FTP-basic, ATP, DDP and ABLAP) as well as at least one more protocol at the session and presentation layers - FTP-extended. FTP-extended implements open, close, read, write, allocate operations. May use a different transport protocol than ATP?

As this mode of operation is one we expect to address later, details of this protocol have not been specified. We may consider emerging protocols such as Microsoft's MSNet. Implementation note, RFP and the transport protocols it uses must reside in the system heap on the Macintosh so that the File Server volume is accessible in all application contexts - this may be very tight on a 128K Macintosh. FTP protocols need not reside there since they are only needed when an ornament or application explicitly uses them - they reside in the application.

## Interconnect Functionality - Direct Connect

When directly connected to a Macintosh, the Personal File provides the clocking on the asynchronous port allowing the very high data rate (987Kbps). The transport protocols will be very simple. They need only support point-to-point, single-port operation with error detection and retry. These will be very similar, if not identical, to those developed in conjunction with Tacmar.

The session/presentation level protocols will be special-cased remote procedure calls that translate the Macintosh OS External File System interface into a byte stream for transmission to the Personal File where they result in remote invocation of the procedure. Results/data are returned to the External File System interface in a similar fashion. Implementation note, both the direct connect transport module and the module mapping External File System calls into requests for remote file operations must reside in the system heap on the Macintosh so that the Personal File volume is accessible in all application contexts.

Additional protocol constructs will be defined for downloading of software and running diagnostics on the FS/PF.

## Hardware Functionality

There are four major hardware components: the chassis and plastic skins, a power supply, a rigid disk drive and a main logic board.

**Chassis and Plastic Skins:** The overall dimensions of the FS/PF will be approximately 360mmW x 105mmH x 240mmD. There is no provision for additional components to be mounted in the box after assembly by Apple.

**Power Supply:** The power supply provides 62 watts at 5v and 12v. This capacity is adequate to power the digital board and any one of the drives currently under consideration for the FS/PF. The only capability for externally accessing this power is via the RS232C modem connector.

**Rigid Disk Drive:** The rigid disk drive is a 5 1/4" unit in the industry-standard, full-height form factor. Several drives are currently under consideration ranging in capacity from 10Mbytes to 67 Mbytes (or possibly two of them producing two variants of the FS/PF).

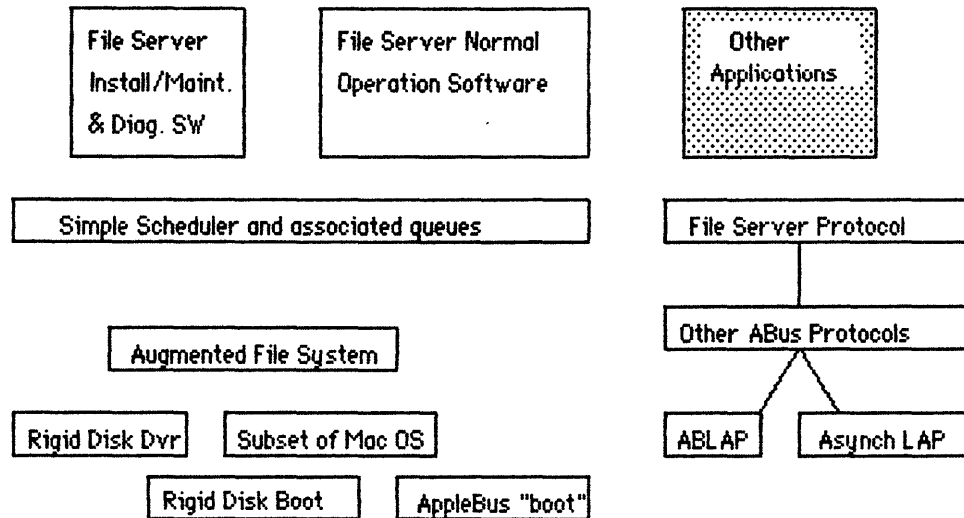
**Main Logic Board:** Wherever possible the main logic board is identical to a Macintosh. Video, sound and other irrelevant functionality has been removed, rigid disk and expansion functionality has been added. The CPU on the board is an M68000 with 128K of RAM and 32K of ROM. The ROM contains those parts of the Macintosh OS relevant to the FS/PF (ie, not Quickdraw) as well as additional code to support booting from the rigid disk and over AppleBus (eg, during installation) and code germane to a file server.

The logic board also supports an AppleBus port, a direct-connect/modem port, interface to an internal rigid disk (either Widget-type interface or ESDI, TBD) and an expansion bus (most probably a subset of SCSI).

While the ROM provides code supporting a Macintosh-style OS and file server functionality, the logic board is a general-purpose computer. Any type of software could be run in this machine to provide a completely different function, eg. seismic data collection and analysis.

# Software Structure

The basic software structure in the FS/PF is shown in the diagram below.



Unless noted otherwise, all software described below is being newly developed for the FS/PF.

**Booting** - The software in ROM supports booting from the internal rigid disk as well as over AppleBus (for installation, maintenance and diagnostics).

**Subset of Mac OS** - The basic operating system functions will be literally those in the Macintosh ROM (plus patches) wherever possible. The Mac OS will be subset to remove extraneous support (Quickdraw, sound, microdisk support) and modified to accomodate hardware differences (clock, timers).

**Rigid Disk Driver** - A Macintosh OS driver to support the internal rigid disk. This will use the VIA/DMA hardware described in the next section and will also perform bad block sparing.

**ABLAP & Other AppleBus Protocols** - These protocols provide reliable transport of data (DDP, ATP) and name lookup services (NBP). These packages are also used in Macintosh and MacWorks software programs.

**Asynchronous LAP** - This package provides access to the RS232 port in the same way the ABLAP package provides access to AppleBus. This allows identical

interconnect functionality to workstations using the File Server remotely via phone lines and modems.

**File Server Protocol** - This package implements the actual filing functions. In basic File Server mode these functions provide file-at-a-time access, in extended mode they provide full file system operations (open, close, read, write).

**Augmented File System** - This file system replaces the Macintosh File Manager (which was not designed to support large disk volumes nor structured file directories). Its design center is for 100+Mbyte volumes containing 10,000+ files. It uses extents (start block, \* blocks) pairs to describe free space and files on the disk. It stores file information in a Btree-structured directory which provides a hierarchically-structured file name space.

**Simple Scheduler and Associated Queues** - Above the system level, the Macintosh OS provides a single-process environment for applications. Since the FS/PF wants to provide the appearance of concurrent activity (multiple workstations storing and retrieving data "at once") there is a Simple Scheduler that provides an environment in which quanta of work can be placed on queues for execution. For reasonably written programs (ones that don't run for a long time before enqueueing further work and returning to the Scheduler) this provides a simple, efficient but not general purpose "multitasking" environment.

**Other Applications** - Other applications can be written and installed on the File Server to execute in conjunction with the File Service software. These applications must be carefully written to observe the constraints of the Scheduler noted above and must also be carefully debugged - there is no memory protection and bugs in these applications will take down the entire system.

The discussion above has addressed only the software running in the FS in normal situations. Additional overview information will be added here describing installation and diagnostic software.

## Hardware Structure

The basic structure of the main logic board is shown in the diagram on the following page. As mentioned earlier, the logic board is very similar to the Macintosh logic board in all areas of common functionality. Video, sound, microdiskette, mouse, keyboard functionality has been removed - functionality to support the rigid disk has been added.

This allows us to leverage of existing system software since significant hardware addresses are identical wherever possible. It also means that many of the parts are already double-sourced and being ordered in volume for Macintosh.

The processor is an 8Mhz 68000 with a 16-bit data bus and 24-bit address bus. The processor runs at a full 8Mhz (less time consumed for RAM refresh) except for cycles taken during disk I/O by rigid disk DMA (see below).

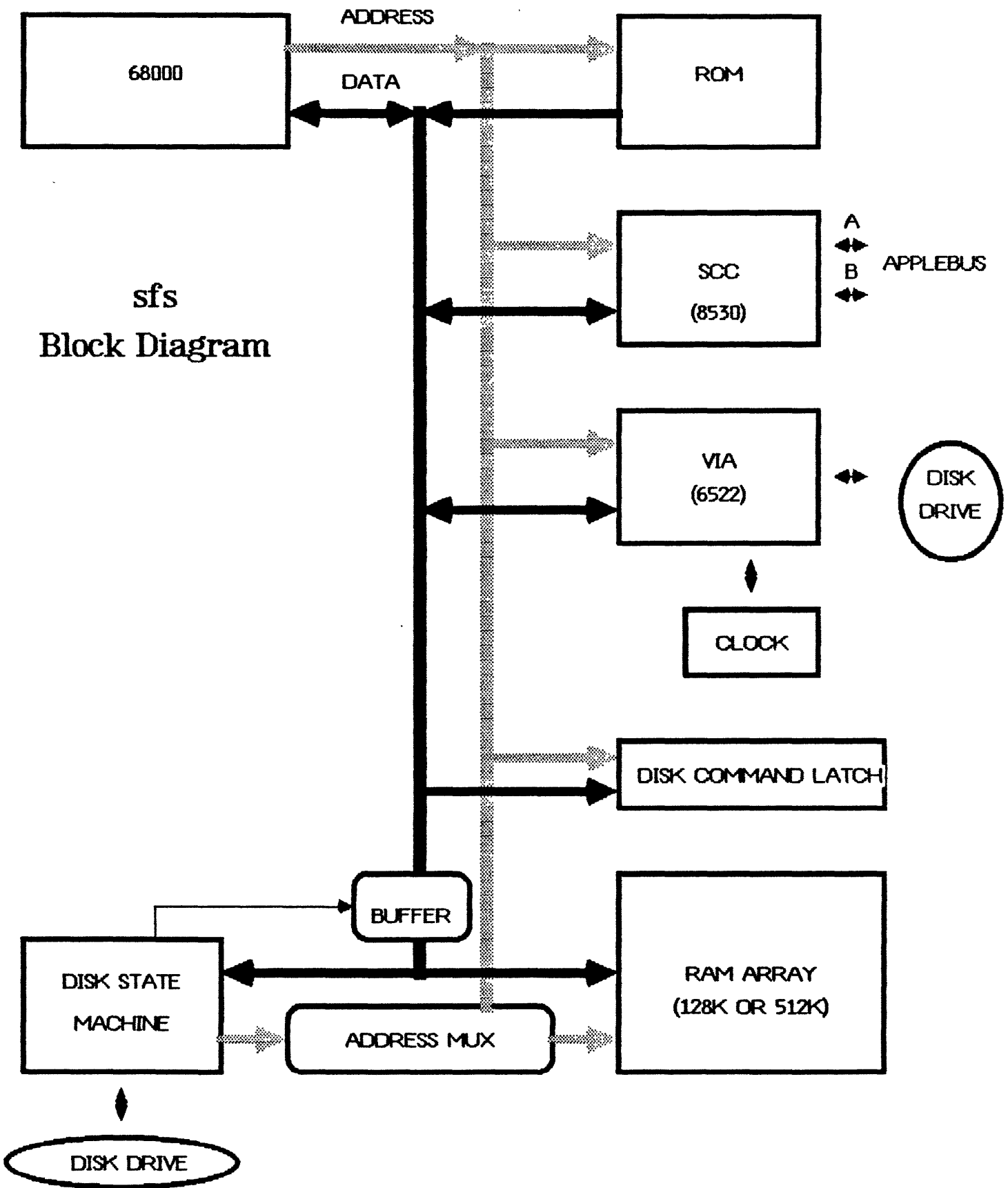
The ROM is 32Kbytes (possibly 64Kbytes) and begins at location 4M (\$40 0000) in the address space. It contains many functions identical to those in the Mac ROM with the deletion of obvious modules such as Quickdraw and the addition of rigid disk and AppleBus booting, disk and AppleBus drivers, etc.

The File Server has 128Kbytes of RAM (expandable to 512Kbytes with 256Kbit RAM chips). After power up the RAM starts at location 0 after power up. The RAM is refreshed by hardware in conjunction with the rigid disk I/O circuitry.

Communications facilities are provided by a Zilog SCC chip support both AppleBus (Rs422) and asynchronous (RS232) communication. (The exact speed that can be supported on the asynch port without losing AppleBus data is a function of software but expected to be at least 9600 baud.) The two SCC ports are externally accessible via DB-9 connectors.

The VIA performs several functions. The clock will be implemented using a very lower power chip and battery such that it will run "forever" without external power. The clock chip also provides 256 bytes of permanent memory.

Rigid disk I/O is performed through several components. The head positioning is done through the VIA. A 2-bit command latch is set to indicate format, write, read and read ID operations. Actual data transfer is performed via direct memory access by the disk state machine and uses the top 1Kbytes of RAM. The address multiplexor and data bus buffer are used to "lockout" the 68000 as necessary



sfs  
Block Diagram

during disk operations (and memory refresh).

At this time the expansion bus for the FS/PF is TBD - its circuitry is not shown on the block diagram. The goal is to use a 25-connector shielded cable - it must meet FCC regulations. The most likely choice is a subset of SCSI (Small Computer Systems Interface) which is being widely used in new peripheral products. Apple parallel and modified ESDI (Extended Small Disk Interface) are also being considered.



## Development and Product Staging

The FS/PF is the first product being addressed in development and will be the first product introduced. It will be followed in 60-90 days by the Tape Backup Unit and Expansion Box, as/if appropriate.

The current plan is to offer the basic FS/PF in 10 and 67Mbyte configurations. The Tape Backup Unit will be the next product to be developed and introduced. The Expansion Box might never be developed - the user would get additional storage by adding another file server, instead of making his existing file server larger - see *Unit Cost*. (If the basic FS/PF is only offered in a 10Mbyte configuration, the Expansion Box will be the second product developed and introduced.)

The development of the user-level functionality will address File Server Mode and Remote File Server Mode first. These are sufficient for an initial product introduction. Additional research is required to determine whether Personal File Mode or Extended File Server (EFS) Mode should be developed next. First product introduction is expected to include File Server functionality and either PF or EFS functionality.



## Issues

- \* Disk size or sizes for the basic FS/PF - 10Mb only; 10 and 67 Mb; 40Mb only
  - \* Expansion bus - SCSI (how many pins?), ESDI (RFI problems, how many pins?), extended Apple parallel, other?
  - \* Can we get 64k of ROM for the same price as 32k given we already buy lots for Mac?
  - \* Should we implement Extended File Server mode before Personal File mode?
  - \* Should we have larger versions of the one box file server or Expansion Boxes?
- Do we want to add storage via additional servers or via Expansion Boxes?