UCSD p-System II.0

INSTALLATION GUIDE

TABLE OF CONTENTS

I. INTRODUCTION

This is the installation manual for the UCSD p-System II.0 on the Altos ACS-8000-15 microcomputer with the RAMDISK feature. The p-System is provided by Softech Microsystems, Inc. The adaptation of the p-System to run on the Altos computer was done by Sorrento Valley Associates and includes some additional performance enchancements. All of the capabilities and attributes of the p-System are passed on to the user unhindered.

I.1 Ramdisk

One of the most important enhancements of the p-System on the Altos is the implementation of the 'ramdisk' feature. The Altos ACS-8000-10,12,14, and 15 series computers contain additional memory on the computer board which the standard p-System does not know about. SVA has enhanced the p-System's peripheral support (BIOS) to utilize this additional storage and use it as a super high speed 'disk'. This disk is used and manipulated like any other disk volume in the p-System. Because this memory storage has no mechanical parts, no latency, and no seek time, access to data in the 'ramdisk' is instantaneous. In addition, the DMA controller is used to transfer the data from the ramdisk to the p-System resulting in a total response time noticeably faster than a hard disk.

The storage available in the ramdisk is relatively small, only 280 blocks. This is about the size of half of a single density floppy disk. This is sufficient however to store an extremely large application program. If the application has many segment procedures, the system will be able to swap them in and out quickly enough to create the appearance that they are always resident. The storage space available on the ramdisk is also large enough to store a few necessary system files with which to develop programs such as SYSTEM.FILER, SYSTEM.EDITOR, SYSTEM.COMPILER, and the application files themselves. Just like any other p-System program, having the system programs resident in the ramdisk results in speedier performance.

Because the ramdisk is in memory, it will not retain data when the computer's power is shut off or if the system is booted. This must be taken into consideration when using the ramdisk.

The ramdisk provides the application programmer many creative possibilities, from high speed program performance to an application running in an unattended mode in which having a floppy disk loaded at all times is undesirable.

I.2 The 9511 Hardware floating point option

Another optional enhancement to the p-System is the support for
the 9511 floating point hardware on the Altos computer. Sorrento
Valley Associates has modified the p-system to do its arithmetic
calculations on the 9511 including the transcendental functions
such as SIN and COS. This hardware offers a speed improvement of
anywhere from 1 to 10 times over the standard software arithmetic
calculations. A p-System using this option is available as part of
this distribution package. It may be used for those applications
which are arithmetically intensive. Since the 9511 has a data
format different from the data format used in the p-System, the two
varieties of p-Systems (Software floating point and Hardware
floating point) are incompatible. A program using hardware
floating point arithmetic cannot be successfully run on the
software floating point system or any other p-System. Therefore,
the program becomes non-portable. To make the program portable, it
is simply a matter of recompiling the program while the Software
floating point system is active.

The HFP and SFP systems are chosen by selecting the proper p-
code interpreter. Refer to the 9511 option installation section in
this manual.

II. BOOTING THE SYSTEM AND USING IT

The following floppy disks are found in the distribution package.

P2RAMSD: Bootable single density System disk, ready to run. Just insert the disk in the right hand drive on the Altos computer and press the boot button. This disk is capable of communicating with the ramdisk and double density disks as well. It contains all the 'SYSTEM' files and some setup utilities for adapting the system to the video terminal you are using as the console device.

P2RAMDD: Bootable double density system disk, ready to run. It is the same as the P2RAMSD: disk described above.

HSE2RAM: This disk contains additional files so that system disks with configurations other than the two provided with the distribution package can be created by you. These include support for the Ramdisk and the 9511 floating point hardware options.

UTILITY: This disk contains all of the utility programs provided as part of the UCSD p-system as well as those provided by Sorrento Valley Associates.

NEWADEX: This is a copy of the ADEX diagnostic disk you have received with your Altos computer with one additional program on it. This program is a utility which allows you to configure and write the bootstrap section on a p-System disk you wish to make bootable.

II.1 <u>PREPARING YOUR SYSTEM FOR NORMAL USE</u>

The distribution system comes ready to run and is all set up for your computer for the most part. There are a few things you must do in order to actually get down to business and use your system. This consists of the following activites:

1) Backup the distribution disks.

2) Configure the p-System for your terminal.

3) Explore and use the p-System if it is unfamiliar to you.

4) Determine if you need to create other system configurations.

II.1.1 Backing up the distribution disks

First you must make backup copies of your disks. This is best done by using the ADEX utility and diagnostic disk provided with your Altos computer. Use the 'COPY' program to make duplicate copies of the single density disks. Use 'DCOPY' to copy the double density disks. Be sure to use the 'COPY ALL' option as some of the disks contain information on the boot tracks (tracks one and two).

Use the FLPYFORM program also found on the ADEX disk to format single and double density floppy disks as required.

Once you have made backup copies of all your disks, boot either the single density or double density system disks.

II.1.2 Terminal setup

Unless you are using a SOROC IQ-120 terminal, you will have to 'setup' the p-System so that it will control your terminal properly. This must be done before going on since proper terminal action is essential to the p-System. Refer to the terminal setup section of the UCSD Pascal Users Guide.

Briefly, setting up the system to communicate with your terminal consists of the following steps. First run the SETUP program provided on the system disk. (This file may be removed later when the setup operation has been completed). Once this program has been successfully used, the operating system will work as it should, with the prompt lines appearing on the upper left hand corner of the screen. In order for the editor to work properly, however, the proper 'GOTOXY' procedure for your terminal must be prepared and compiled as described in the documentation, and then

bound into the file SYSTEM.PASCAL with the BINDER utility. Once these two steps have been completed, the system will be configured completely.

Since the screen oriented editor requires a correct 'gotoxy' procedure in order to operate correctly, initially the editor is not available to create the file you need to make the editor work. In this case you should use the line oriented editor called YALOE. This editor will work with any terminal including teleprinters. Refer to the UCSD p-System user's guide for instructions on using this editor.

At this point, if you have never used the p-System before, it is best to 'putter around' and become familiar with the programs, utilities, and their uses.

Later on, after you have used the system and have read thoroughly through the installation manual, you may determine you wish to create system disks which implement some of the optional configurations. These options are described in the appropriate section of the manual.

II.2. <u>UTILITY PROGRAMS</u>

The following are special utility programs provided by Sorrento Valley Associates. All are found on p-System disks unless otherwise specified.

P2RAMSYS - Sets the default configuration of the p-System and writes the bootstrap routine to the bootstrap section of a system disk. This program is found on the NEWADEX disk.

SYSTEM.STARTUP - A program which prompts the user for the date and sets the default prefix volume. The source is provided so that a custom system.startup program may be built by the user.

CONFIGURE - Allows the user to change the default configuration of the system. These changes affect the density of the floppy disk drives, whether the printer port is serial or parallel, and the baud rate of the serial port.

DISKCOPIER - A p-System floppy disk copy utility which also compares the new copy with the old providing additional copying confidence. The COPY and DCOPY programs found on the ADEX diagnostic disk can also be used to copy p-System disks.

TESTXFP - A utility which will test the software or hardware floating point arithmetic sections of the system and optionally allow you to make relative timing comparisons.

II.2.1 The bootwriter utility

To make a bootable p-System disk it is simply a matter of initializing a volume in the standard manner, putting the appropriate 'SYSTEM' files on the disk, and writing the bootstrap onto the boot tracks (first two tracks on a p-System disk).

The appropriate set of system files MUST include the following:

SYSTEM.PASCAL - the operating system with the GOTOXY procedure in it
SYSTEM.BIOS - communicates with all I/O devices
SYSTEM.MICRO - the p-code interpreter, changes with options desired
SYSTEM.MISCINFO - info on your terminal, date, etc.

Of course, you will need other files in order to do anything useful but you will have to determine which they are. The above files comprise the 'run time system' and are essential to any activity.

Refer to the UCSD p-System users manual for details on how to initialize a volume using the Filer.

Refer to the following instructions to create a bootstrap on a system disk you wish to make bootable.

1) Insert the NEWADEX disk and boot it. Run the program called P2RAMSYS.

2) Answer the questions which will set up the default configuration for the system when it is booted. For more information on what the configuration is all about, refer to the section describing the CONFIGURE utility.

3) Insert the disk you wish to make bootable into the drive specified by the bootwriter program and answer yes to the prompt.

You now have a bootable system disk. When the system is booted, a default configuration will be set up. This configuration can be changed at any time with the CONFIGURE utility. See that section in this manual for a complete description on how to run this program and what it does to the system.

II.2.2 The SYSTEM.STARTUP utility

The SYSTEM.STARTUP file is a special p-System code file meant to be executed at all times thereby creating a 'turnkey' system. The users application program, when it is fully prepared, may be renamed to SYSTEM.STARTUP. When the system boots up, or when the system recovers from a runtime or execution error, the system will always start the application (SYSTEM.STARTUP) over again. In this manner the user never sees the UCSD p-System promplines and doesn't need to know how to use the operating system in order to work with the application program.

The SYSTEM.STARTUP program provided in the distribution package, however, does not operate in the same manner and in fact executes only once at boot time. This program simply enhances the boot operation by prompting the user to enter any changes to the date and automatically sets a default 'P)refix'. This makes it unnecessary to go into the filer to do these common operations.

When the system is booted, it displays the regular p-System greeting, displays the current date, prompts the user for a new date and positions the cursor below the current date. If the actual date is the same, the user may hit return and the system will continue. If the date is different, the user need only enter the part of the date which is different. Once the date has either been accepted or is modified the system will set the P)refix to volume #5:. This means that if you don't specify a volume number when referencing a file, the system will use the default prefix of #5:. The user may modify the startup program to set the prefix to any other disk volume.

The source to the startup program is included and may be modified as desired or used as a base from which to build your own custom SYSTEM.STARTUP file.

The startup file provided is not a true startup file since it is meant to operate only once. You have a choice of having your startup program execute once at boot time, or all the time. Refer to the source listing to learn how this program operates.

II.2.3 The Configure utility

The configure utility allows the user to change the setting of
peripheral device operating parameters. These parameters include
the following:

* Floppy density

* Serial or parallel printer operation

* Baud rate of the serial port

The p-System supports 12 volumes.   These volumes may be the
console device, the disks, the printer, the remote port, the
graphics unit, etc.

```
#2:      CONSOLE:
#4:      right hand floppy: boot density
#5:      left  hand floppy: boot density
#6:      PRINTER:  serial or parallel
#7:      REMIN:    serial port only if printer is parallel
#8:      REMOUT:   serial port only if printer is parallel
#9:      ramdisk device if 9 volumes are enabled
#11:     right hand floppy: 'other' density
#12:     left  hand floppy: 'other' density
```

The volumes which do not appear above are not utilized in this
system. The volume assignments are the same as the standard p-
System assignments.

II.2.3.1 Disk density

When the system boots up, a default density is selected. This
is, of course, the same density as the p-System disk being booted
from.   The configure program will allow the same drives to work
with floppies of the 'other' density by using the configure
program.   The configure program enables 8, 9, or 12 volumes
(total).   This simply means that if 8 volumes are selected, the
system does not know about volumes above #8:. If 9 volumes are
selected, the system will be able to work with the ramdisk but
still only be able to work with disks that are the same density as
the boot density. If 12 volumes are selected, the system will be
able to work with disks of the opposite density from the boot
density.

For example, if you wished to make a double density copy of a
single density disk because you ran out of room, run the configure
utility and select 12 volumes (if it isn't already). Insert the
new (already formatted) double density disk in the left hand drive.
In the filer, Z)ero #12: and specify a size of 900 blocks. This

will create a directory on the double density disk. Then T)ransfer
#4:=,#12:$ and the system will transfer each file individually over
to the double density disk.

In essence, the same two drives are used for either single or
double density operation, each density and drive combination having
a specific volume number.

When the system looks at a volume and expects a certain density
and a disk of the other density is installed, the system will give
the following error message:

PERMANENT DISK ERROR #10

For example, if you were to put a double density disk in the
left hand drive and gave the filer command L)dir #5:, the system
would have set the disk drive hardware into the single density mode
and when it tried to read the disk, found it could not make head or
tail of what was in the drive, and gave the error message. In this
case the user should have specified #12:.

Under certain conditions the operating system will do a volumes
scan. During a volumes scan the system will go through each of the
disk volumes and try to read the directory, either to see which
volumes are online (in the case of the V)olume command in the
filer), or to search the directories for a file you have specified.
A volume scan is also done when you are Z)ero-ing a volume. If the
current configuration has 12 volumes enabled and the system does a
volumes scan, it will always report a PERMANENT DISK ERROR #10
because it will attempt to read a volume of the other density for
each of the physical floppy drives. For example, the system will
try to read the directory of volumes #4: and #11: and read the disk
in the right hand drive twice. One of those times the density will
be wrong and the error message will be displayed. In this case, the
error message should be ignored.

Other error messages that can be generated by the hardware
support environment (BIOS) are listed in the system maintenance
section of this manual.

As a matter of convenience it may be best to set the default
number of volumes to 9 and use the configure program to change it
only when you need to have DUAL density operation.

II.2.3.2 Baud rates

The configure utility will set the baud rate of either the
console port or the auxiliary port. Simply select the appropriate
baud rate from the list displayed. There are no other possible
baud rates other than those given.

II.2.3.3 Printer port selection

It is possible to have either a serial or parallel printer port. Simply make the appropriate selection in the configure utility. If a parallel printer option is selected, then the auxiliary serial port on the Altos computer can be used for the p-System's remote port thereby enabling the REMIN: and REMOUT: volumes.

If a serial printer port is selected, it will operate at the baud rate specified when the baud rate was selected, however, the REMIN: and REMOUT: volumes become disabled and the parallel port on the computer becomes unusable.

II.2.3.4 Disk volume sizes

When initializing a disk volume with the Z)ero command in the filer, specify the following block sizes:

* 494 blocks - single density floppy
            (uses IBM 3740 format: 128 bytes per sector
                                    26 sectors per track
                                    77 tracks)

* 900 blocks - double density floppy
            (uses Altos format: 128 bytes per sector
                                    48 sectors per track
                                    77 tracks)

* 280 blocks - ramdisk
            (uses SVA format: 512 bytes per sector
                                    288 sectors per track
                                    1 track)

II.2.4 The Diskcopier utility

The diskcopier utility is a general purpose disk copying
program. It has the following features:

  * Copy between either single density or double density floppies

  * Copy - boot tracks only
           - p-System section of disk
           _ all of the disk

  * Copy with comparison to original

The diskcopier utility is provided in addition to the COPY and
DCOPY utilities found on the ADEX diagnostic disk. Both work
equally well, however, the DISKCOPIER utility will copy and then
compare the copy to the original to check for accuracy. The COPY
and DCOPY programs do not do this.

To copy double density disks with DISKCOPIER you must have
booted a double density p-System or have set the configure to 12
volumes.   In the latter case, you would copy between units 11 and
12. These copy utilities copy disk IMAGES, unchanged, between the
source and destination disks.   Therefore,   the source   and
destination disks must be the same density. If you wish to copy
only certain files, or transfer all the files from a disk of one
density to a disk of the other density, you must use the Filer to
do that.

As is the case with all the disk copying utilities, the source
and destination disks must be of the same density. The disks must
be properly formatted.   The disks need not have been 'Z)ero-ed'
with the Filer.

II.2.5 The TESTXFP utility

The TESTXFP utility tests the mathematical operations of the system. In addition it can provide some helpful timing information to assist in determining the system throughput. Actually there are two versions of the TESTXFP program. They are as follows:

TESTSFP - Tests the software floating point system and can only be run on a software floating point p-System.

TESTHFP - Tests the hardware floating point system and can only be run on a p-System with the hardware floating point device (9511) installed.

The program is self-prompting and easy to use. No further explanation need be given. The following is a comparison of some selected mathematical operations done on the SFP and HFP systems. The times given below are relative numbers.

operation	SFP	HFP
dummy	.557	.555
X * Y	1.642	.912
X / Y	1.912	.923
X + Y	.935	.847
X - Y	.927	.880
SIN(X)	12.885	2.526
SQRT(X)	13.058	1.120
X <> Y	.775	.887
X > Y	.778	.893
X >= Y	.798	.912

Note that significant speed advantage is gained in the HFP systems when transcendental operations are being done such as SIN and SQRT. However, it takes just as long to do other math operations such as add, subtract, and comparison operations. Multiply and divide offer almost a factor of 2 speed improvement. The decision to use the HFP p-System should be carefully considered as the resultant p-Code file is not compatible with the SFP system supplied, or any other p-System. However, to make the code file compatible with the standard p-System it is simply a matter of recompiling the program while the SFP system is active. The mathematical operations of the compatible application program will of course run more slowly compared to the HFP compiled application.

III.   RAMDISK SETUP AND OPERATION

The ramdisk feature provides many new opportunities with respect to system operation and performance.   The following sections provide important information concerning proper setup and usage of the ramdisk. Please read these sections carefully.   The ramdisk provides improved disk access speeds but important consideration must be made to the volatile nature of the storage. Various options in the use of the ramdisk will be mentioned.

There are some important things to know about the volatility of the ramdisk. Since the ramdisk is purely a semiconductor memory device, it retains all the attributes of such memory.

* The ramdisk will forget its contents if there is a power failure, or power glitch of sufficient duration.

* Pressing the reset button disrupts the refreshing of the memory devices and data will be lost ranging from a few scattered bytes to extensive loss of memory contents.

* The fact that you can read the directory of the ramdisk after rebooting the system does not mean that the file contents are okay.

* During normal system operation, the reliability of the ramdisk storage is very high.   If your program can exectute in memory indefinitely without failure, then the ramdisk storage will   exhibit the same reliability.   It is recommended, however, that if your application runs unattended for long periods of time, occasional error checking should be done and a restart capability should be implemented.

III.1 Initializing the ramdisk

The ramdisk should be used just like any other disk volume in the system. There are no special utilities or operations which deal with the ramdisk. Initialization of the ramdisk is performed just as it would be with a floppy. In the filer, the 'Z)ero' command is used to put a directory on the volume. The ramdisk is permanently setup to respond as volume #9. The size of the volume to specify is 280 blocks. The specification of a duplicate directory is not really useful but is optional. (However, the use of a duplicate directory on floppy and hard disk volumes is stongly recommended.)

After the volume has been initialized, list its directory to be sure it is actually there. If it is not, be sure you have enabled at least 9 volumes with either the configure utility or have enabled 9 volumes as part of the bootstrap setup.

Transfer the desired files to the ramdisk normally. It is suggested that the following sections be read so that you will have a better idea of how you may wish to use the ramdisk and consequently which files you'll be needing.

III.2 Uses of the ramdisk

This section discusses the possible uses of the ramdisk with
regard to the operating system and the way it works with the
ramdisk. A few terms will be redefined here though they are also
defined in the UCSD Pascal User's Guide.

PREFIXED VOLUME - The user may set the default 'P)refix' in the
filer. This is simply the volume number to use when the user does
not specify one. The default volume is #4. The system simply fills
in the volume number for you in a file specification without you
having to type it in. If your application program opens a file
without specifying a specific volume number, the volume number set
by the prefix command will be used.

SEGMENT PROCEDURE - The p-System has a feature in which a large
program may be fit into a small memory space. This is done with an
'overlay' technique. Your application may be divided into several
segment procedures. When one of these procedures is called, it is
loaded into memory from the disk into an overlayable area. When
another segment procedure is called, it will be read into memory on
top of the old segment procedure which is not needed anymore. This
technique is simple to implement and handled automatically by the
operating system. If your program has many segment procedures or
jumps between a couple of segment procedures regularly, many disk
accesses will be made. For a floppy based program, this will
result in a considerable slowdown in program speed. If the program
is resident in the ramdisk, however, the disk accesses will be very
rapid and will result in a greatly reduced performance loss, if
any.

ROOT VOLUME - This refers to the volume which contains the file
SYSTEM.PASCAL. This is the operating system file. If this file is
moved on that volume, the system must be rebooted (e.g. a K)runch
operation in the filer). When moving between the editor and filer
or other operating system programs, or when starting or stopping
the execution of your program, the root volume will be accessed and
the appropriate segment procedures of the operating system will be
accessed and loaded into memory. It is possible to select one of
two possible volumes as the root volume; the floppy disk as volume
#4 or the ramdisk as volume #9. Selection and use of these options
will be explained later.

SYSTEM.MICRO - This is the p-code interpreter. This program
along with SYSTEM.BIOS are the only machine language code files
(e.g. Z80 instructions). All the rest of the files are p-code files
and come direct from Softech without modification. However, the
BIOS code was written completely by SVA and the interpreter was
modified by SVA to implement the 9511 option, the ramdisk option,
and the root volume option. A disk in the distribution package
contains various files to be used as SYSTEM.MICRO. The choice of
the proper p-code interpreter will be explained later.

III.2.1 Ramdisk as the prefixed volume

If you have set the prefix to be #9: with the filer, then all
the files you specify will be gotten from the ramdisk, unless you
have specifically used a volume number or name in the file
specification. Putting your application program on the ramdisk
will result in improved performance only if it has segment
procedures or if it opens and uses files on the ramdisk.

III.2.2 Ramdisk with system files

Putting system files on the ramdisk such as SYSTEM.COMPILER, EDITOR, LINKER, ASSMBLER, etc., will result in improved performance of those programs. When the p-System boots up, it scans all the volumes online and finds out on which volume each of the system programs reside. The best way to make the system programs ramdisk based is to rename them on the floppy volume or not have them on the floppy volume at all. Then, after the ramdisk has been Z)ero-ed, copy the desired system files from a floppy source to the ramdisk. There must only be one copy of each program 'online'. The first time you use one of the system programs, the operating system will look on the root volume for it and when it doesn't find it it will search all volumes online until it finds it. From then on, the system will go directly to that volume each time the program is invoked.

The system programs will operate more quickly when their segment procedures are accessed. In the case of the compiler in the swapping mode, this will result in speedier program compilation, especially if the program to be compiled is also on the ramdisk.

III.2.3 Ramdisk as the system unit

This section discusses the effects of making the ramdisk the system volume or root volume. If the ramdisk is the root volume, then the entire operating system will operate much more quickly. There is virtually no delay experienced when exiting from the filer to the operating system's command level.

Two p-code interpreters are provided which have volume #9 (ramdisk) set as the root volume. There are two more which have volume #4 set as the root volume. The system is distributed with the root volume set as #4: and with the software floating point option. The p-code interpreter files are the following:

RAM.HFP.V9.INT - hardware floating point, ramdisk is root volume.

RAM.SFP.V9.INT - software floating point, ramdisk is root volume.

RAM.HFP.V4.INT - hardware floating point, floppy is root volume.

RAM.SFP.V4.INT - software floating point, floppy is root volume.

The appropriate interpreter to be used is transferred to the root volume and renamed SYSTEM.MICRO. When the system boots it always loads the SYSTEM.BIOS, and SYSTEM.MICRO from the floppy disk. If the root volume is #4:, then operating system (named SYSTEM.PASCAL) will be found on the floppy. If the root volume is #9: then the operating system will be found on the ramdisk.

Note that if the root volume is #9:, the ramdisk must be previously set up containing the appropriate set of system files, including SYSTEM.PASCAL and SYSTEM.MISCINFO. SYSTEM.MICRO, and SYSTEM.BIOS need never be loaded to the ramdisk. These two files are loaded into memory at boot time and their files on the disk are not used during system operation.

The easiest way to create a ramdisk based p-System is to first boot up a floppy based system, initialize the ramdisk, transfer the appropriate set of system files to it, then reboot using a special floppy with the following contents:

SYSTEM.BIOS
SYSTEM.MICRO
The bootstrap properly written to the disk.

Remember that pressing the boot button interferes with the ramdisk memory refresh operation and the good possibility exists that the integrity of the ramdisk storage will be harmed. This little experiment should be tried, however, to illustrate how a ramdisk based operating system performs and must be setup.

III.3 Automatic initialization of the ramdisk

Due to the wide variety of ways in which the ramdisk may be used and setup, it is up to the application programmer to determine the best way to automatically initialize the ramdisk for a turnkey system. The following suggestion illustrates how this may be done.

1) Setup the ramdisk as desired.

2) Execute a program you have written which will read the 'image' of the ramdisk and write it to a file using BLOCKREAD and BLOCKWRITE. You need only save as much of the ramdisk as is used, i.e. all 280 blocks need not be saved if the end of the ramdisk has nothing stored in it.

3) Put additional code in your SYSTEM.STARTUP file to transfer the ramdisk 'image' from the floppy file to volume #9:. The ramdisk will now have the files you want in it.

4) If it is desired to have the ramdisk as the root volume, the user must be prompted to insert a new boot disk setup as described previously.

5) The system must be rebooted without pressing the boot button. This can be done by reading the first sector on the first track and loading that sector to memory location 0000, setting the Z80 interrupt page to page 0000 and then jumping to location 0000. When the sector is transferred to this location, the p-System will not operate any longer requiring that this boot operation be a machine language program which may not return to the p-System.

IV. ADDITIONAL FEATURES AND FACILITIES

The following section describes some additonal features and facilities available and how to implement the ramdisk and floating point arithmetic options.

IV.1 Ramdisk option selection

Ramdisk options include the selection of the appropriate root volume. It is assumed that you have read the ramdisk section completely. Simply select one of the following interpreter files, put that file on the booting volume and rename it to SYSTEM.MICRO. The new system will become active the next time the system is booted.

RAM.HFP.V9.INT - hardware floating point, ramdisk is root volume.

RAM.SFP.V9.INT - software floating point, ramdisk is root volume.

RAM.HFP.V4.INT - hardware floating point, floppy is root volume.

RAM.SFP.V4.INT - software floating point, floppy is root volume.

IV.2 Hardware / Software floating point option selection

   To select the appropriate floating point option, choose one of
the following files, transfer it to the boot volume, and rename it
to SYSTEM.MICRO. The new system will become active the next time
the system is booted. The 9511 arithmetic hardware must be
installed or the system will hang.

RAM.HFP.V9.INT - hardware floating point, ramdisk is root volume.

RAM.SFP.V9.INT - software floating point, ramdisk is root volume.

RAM.HFP.V4.INT - hardware floating point, floppy is root volume.

RAM.SFP.V4.INT - software floating point, floppy is root volume.

IV.3 Peripheral notes

The following pertains to additional features or notes not documented in the UCSD Pascal User's manual.

PRINTER FORM FEED - typing a <control>T at the terminal keyboard will cause a form feed character to be sent to the printer port.

PRINTER MONITOR - Typing a <control>P will cause everything written to the screen to also come out on the printer. Typing another <control>P will turn off this action.

REAL TIME CLOCK - The p-System's real time clock is supported and the Pascal function TIME(HIWORD,LOWORD) will return a number which is incremented every 60th of a second. The user should note that the clock frequency is not exactly 60 Hz and a small test program should be written to determine the exact frequency of the clock if this must be known. Typically the clock frequency is 61.03 Hz.

V. SYSTEM MAINTENANCE

If you are having trouble with your p-System, read the following to determine who you should call.

APPLICATION PROGRAM OPERATION - If you get error messages or the system crashes while you are running a special application program, call the author of that program.

TROUBLE BOOTING, PERIPHERAL DEVICE COMMUNICATION PROBLEMS. After you have determined that the peripheral devices are operating properly and that the configuration parameters have been properly set, you may call Customer Service at Sorrento Valley Associates (714) 452-0101.

COMPUTER HARDWARE TROUBLE - Call Altos computer systems. See the section on diagnostic tools for information on determining hardware troubles.

P-SYSTEM PROGRAM OPERATION - If you are having trouble understanding something or the system is giving you unexpected results, call the SUPPORT group at Softech Microsytems in San Diego.

V.1 Diagnostic tools

    The NEWADEX disk supplied with the distribution package as well as the original ADEX disk received with your Altos computer have the following diagnostic programs which are of interest in maintaining your computer hardware.

    MEMTEST - tests all memory including ramdisk memory

    FLPYTEST - tests all aspects of floppy disk operation.