SECTION VII

Software

SOFTERM INSTALLATION

The following procedures will guide you through the installation of SOFTERM on your IBM PC or equivalent system.

1. Boot the system.

2. Put the ISI SOFTERM Installation Diskette into Drive A.

3. At the DOS prompt, type A:AMINSTAL
   After all of the files have been transferred, SOFTERM is installed.

SOFTERM DOCUMENTATION

SOFTERM is a software package which will enable your IBM PC or IBM PC look-alike to run CAIR Rating or SuperVUE on the Alpha Micro Computer. CAIR Rating and SuperVUE require specific terminal types in order to run correctly. SOFTERM makes it possible to emulate these terminal types. Your IBM PC or equivalent will be emulated to the Televideo 950. Please refer to the SOFTERM PC manual, Chapter 6, page 109-111 for the Televideo 950 Keyboard Functions.

There are two ways in which your IBM PC or equivalent may be used with the Alpha Micro. The first is by modem. The second is a direct connection by cable.

A. MODEM

1. Initiating SOFTERM
   A. At DOS prompt, type ALPHAM
   B. Hit CARRIAGE RETURN on the Setup Options Menu to accept the Online Operations option.
   C. Dial the modem as you normally would.
      Remember that the CAPS LOCK should be set.
      Note: if the security monitor is on you must enter a password.
   D. You should now be on the Alpha Micro System at the AMOS command level.
      Type TRMTYP TELVDO
   E. SOFTERM is now set and you may begin work.
      To run the System Directory, type BYE.

2. Exiting SOFTERM and hanging up modem
   A. Get to the AMOS command level.
   B. Enter ALT + "2" to get to the Goto Functions Menu.
   C. Enter F8 to exit SOFTERM and hang up the modem.
      You should now be at the DOS prompt.

B. DIRECT CABLE

1. Initiating SOFTERM
   A. At DOS prompt, type ALPHA
   B. Hit CARRIAGE RETURN on the Setup Options Menu to accept the Online Operations option.
   C Get the AMOS prompt by hitting CARRIAGE RETURN.
      SOFTERM is now set and you may begin work.
      To run the System Directory, type BYE.

2. Exiting SOFTERM
   A. Get to the AMOS command level.
   B. Enter ALT + "2" to get to the Goto Functions Menu.
   C. Enter F8 to exit even if not using the modem.
      You should now be at the DOS prompt.

February 15, 1985

Files Associated With the Prospect System And Their Locations

DSK0:[2,2] -(CMD:)

PS.CMD	- command file to enter PS
SYSEND.CMD	- command file to exit PS

DSK0:[7,6] -(BAS:)

NAMSBR.LIT	- M/L subroutines used by PS
FILSPC.DAT	- data file, list of files used by PS
FILSPC.RUN	- maintenance prog for FILSPC.DAT
NAMES.RUN	- central PS program
NMXPND.RUN	- PS file expand/reduction program
RPGPH1.RUN	- 1st report generator program
RPGPH2.RUN	- 2nd report generator program

XXXn:[22,0] - Prospect System run-time account

PRSPCT.IMG	- image file, a 'picture' of the editing screen.
PRSPCT.PAR	- parameter file, tells PS where fields are located in editing screen and how data is stored in the data record.
PRSPCT.DAT	- actual prospect data file.
PRSPCT.IDX	- index to the prospect data file.
PSBACK.CMD	- specific command file to back up this account.
PSLOAD.CMD	- specific command file to restore this account.

cache

FUNCTION:

Allows you to control the Disk Cache Buffer system, and to find out information about it. Disk "caching" speeds up your system by reducing disk access.

HINTS/RESTRICTIONS:

When used from an operator's account [1,2] or from [1,4], CACHE puts you in an interactive mode in which you can modify the cache.

When used from any other account, CACHE displays statistics about the disk cache system.

For more information on setting up and using the cache, see "The Disk Cache Buffer Manager" in the AMOS/L System Operator's Guide, DSS-10002-00.

FORMAT:

.CACHE RET or:

.CACHE RET
*command RET

OPERATION:

When used from any other account than [1,2] or [1,4], the CACHE command will display something like this:

```
.CACHE RET

Total buffers in cache:      188
Number of locked buffers:    39
Number of unlocked buffers:  149

Total disk read requests:    116409
Number of reads from cache:  82943
Number of reads from disk:   33466

Efficiency:                  71.3%

Disk cache:                  ON

Lock restriction:            OFF
Read restriction:            OFF
Random restriction:          OFF
Dynamic MFD locking:         ON
Dynamic UFD locking:         ON
```

When used from [1,2] or [1,4], CACHE enters an interactive mode. The format for an interactive command in this mode is:

*command{switch} {device}{filename} RET

The commands are:

```
ON     - Turn on the disk cache system
OFF    - Turn off the disk cache system
LOCK   - Lock the specified block or file into the cache
UNLOCK - Unlock the specified block or file
CLEAR  - Remove the specified block or file from the cache
STATUS - Display statistics about the cache
HELP   - List the available commands
LIST   - List static, dynamic, or hash blocks (you must
         specify a switch)
EXIT   - Return to AMOS
```

The switches are:

```
/FILE    - Perform the operation on the specified file
/MFD     - Perform the operation on the specified MFD
/UFD     - Perform the operation on the specified UFD
/STATIC  - List the blocks in the static queue (LIST only)
/DYNAMIC - List the blocks in the dynamic queue (LIST only)
/HASH    - List the blocks in the hash queue (LIST only)
/BLOCK   - Perform the operation on a specified block
```

The switches may be abbreviated. Some examples:

```
*ON RET
*LOCK/FILE PAYROL.DAT RET
*UNLOCK/F  PAYROL.DAT RET
*LOCK/MFD DSK0:, HWK1: RET
*CLEAR/FILE DSK1:PAYROL.DAT[126,10] RET
*STATUS RET
[  Status display is shown, as above  ]
*HELP RET
[  A screen of helpful information is displayed  ]
*LIST/S RET
[  A list of the static queue blocks is displayed  ]
*EXIT RET
```

You may add a CACHE command to your System Initialization Command File (AMOSL.INI) after the final SYSTEM command so that the CACHE is set up right from the time of system bootup. Here is a sample:

```
.
SYSTEM DCACHE.SYS/N/M/U 100K
SYSTEM
.
MOUNT DSK0:
.
CACHE
OFF
ON                          ; Prepare the cache
LOCK/MFD DSK0:              ; Lock Master File Directory
LOCK/UFD [1,4],[1,6]        ; Lock User File Directories
LOCK/FILE VUE.LIT, M68.LIT, SYS.UNV[7,7], SYSSYM.UNV[7,7]
EXIT
.
```

ERRORS:

?Argument error

You specified an invalid argument in the input line. An arrow (^) points to the incorrect part of the line.

?Command error

You specified an invalid command name in the input line. An arrow (^) points to the incorrect command.

?Disk cache does not exist

The disk cache file, DCACHE.SYS, does not exist, is in the wrong account, or was not loaded into system memory by the AMOSL.INI file. Check to see that DCACHE.SYS exists in DSK0:[1,4], and that is in loaded properly.

?Not enough cache space available

There are not enough disk buffers available in the cache to lock the MFD, UFD, file, or block you specified. You may have to allocate more memory to the disk cache system.

?Specification error

You made a syntax error in the input line. An arrow (^) points to the mistake.

?Switch error

You specified an invalid switch. An arrow (^) points to the switch. See the operation section above for the switches.

CHARACTERISTICS:

Allows you to modify the Disk Cache Buffer System, and to display information about it.

CACHE is re-entrant and re-usable.

You must be in a system operator's account [1,2], or in a system account, [1,4], to use CACHE to modify the cache. In all other accounts, it will display a cache status report.

Part Number ZAT-01001-00

XCERT 1.1(115) User Instructions

Introduction

XCERT is a format program that will also allow you to recapture lost defect map information. We recommend that this program be used in lieu of CRT420 for the 5 1/4 inch peripheral disk drives that are sold and supported by Alpha Micro. Please remember that this program will only perform properly if used correctly and will have negative results otherwise.

Alpha Micro's recommended procedure for restoring BADBLK.SYS is to immediately make a copy of the original BADBLK.SYS and then save it in such a manner that a copy may be restored or viewed later. This should be done every time BADBLK.SYS is appended for any reason. Information on saving BADBLK.SYS is available in the AMTS Journal article 1.1.16 published July 1984.

If you do not have a copy of the original BADBLK.SYS file which contains the defect map of your particular drive, it will be necessary to obtain that information before running XCERT. This information is contained on a label(s) attached to the outside of the head/disk assembly of the disk drive. To read this information from the label, it will be necessary to open the system and remove all obstacles, including controller, that obstruct your ability to view the label clearly. In addition, further defects for certain drives may have been added to your drive. This may be done by the user or a technician. Please check the original BADBLK.SYS for this even if you wish to run XCERT.

Operating System Requirements

This program requires AMOS/L version 1.2a or 1.3 to operate correctly.

Program Limitations

There are several limitations to the XCERT program of which the user should be aware.

1 You should not execute XCERT on the device from which you booted the system if at all possible. If this is necessary, you should have XCERT in a warm boot monitor with the correct disk driver.

2 To use XCERT with the print only option, you do not have to mount the drive. However, the device must be defined in the system as a valid device. The print only option leaves the device mounted and any attempt to access it or unmount it will cause the system to hang. If you wish to use this option from an AM-1000, either address the device as the second physical unit in the software or as a disk subsystem.

3 To execute XCERT on any disk drive, it must be defined in the in the system as a valid device

Switch Options

/P This will only prompt you for the defect information and then list the bad blocks related to the defects entered. The drive does NOT have to be mounted to use this option.

/S This indicates that the drive you are certifying is a subsystem. This option not necessary for this version of XCERT.

/L This tells XCERT to make an ASCII BADBLK.LST file of BADBLK.SYS.

/B This will allow the user to type in defects and create a new BADBLK.SYS without formatting the drive. The directory must be intact. The disk drive must be remounted before copying any software to it after using this option and then run SYSACT to clear the drive before restoring any software to it.

/? This will not run XCERT but display on your terminal a list of switches. A unrecognized switch will cause a list of switches to be printed out.

Using XCERT

To use XCERT, type XCERT followed by the device to be formatted. The drive defined must be the first logical of a physical drive and must be mounted. The only exception to this is the print only (/P) option.

Example: XCERT QTM0:{/switch}

Switches must be placed after the device name such as in the following example:

```
XCERT QTM0:/B/L
```

XCERT will then ask you which type of drive you want to format. Type in the appropriate letter. There are currently 5 drive types available. They are:

(A) CMI 10 Mbytes
(B) CDC 30 Mbytes
(C) CMI 30 Mbytes
(D) Quantum 30 Mbytes
(E) Vertex 55 Mbytes

XCERT will then ask you for the serial number of the drive. You are allowed up to 10 characters.

Lastly XCERT will ask you to enter the defects. You will be prompted by a 0: . Enter the cylinder, head, and byte offset followed by a return. You will then be prompted by a 1: The prompt will increase for each defect you enter. Entering just a return without entering any defect information will terminate entry of defects and signal XCERT to format the drive and build BADBLK.SYS.

If you make a mistake entering defects, XCERT allows you to edit any of your entries. A back slash, (\) will back you up to the previous entry; entering a new cylinder, head, byte offset will then write over the previous entry. Typing a number followed by a colon, (5:), will allow you to modify that entry. Typing just a colon, (:), will print out the latest set of defects that you have given XCERT. Typing a question mark, (?), will print out a brief summary of the editing commands and retype the current list of defects. The rubout and ^U will also work for the current input line The following is an example:

```
0: 123,3,323
1: 32,0,90
2: 200,1,2133
3: \                    \   entered causes a prompt for
                            previous entry
2: 200,0,2133;              This replaces the previous entry
                            for 2
3: 212,1,111
4: 101,4,1101
5: 1:                   1:  takes you back to entry 1
1: 12,0,90                  After over writing the 1 entry
                            you are returned to where you
                            were
5: 200,1,30
6: 201,3,122
7: 311,1,2122
8: :                    :   causes the current listing to
                            be printed out
```

```
0: 123,3,323
1: 12,0,90
2: 200,0,2133
3: 212,1,111
4: 101,4,1101
5: 200,1,30
6: 201,3,122
7: 311,1,2122
8:
```

A carriage return terminates entry of defects

XCERT checks to see that you enter a valid cylinder, head and byte offset and tells you if you do not. If you enter just a cylinder number everything else will default to zero.

After XCERT formats the disk and builds BADBLK.SYS, it will tell you how many bad block entries it has made. This may be more or less than the number you entered. Because defects may lie on sector boundaries, it might require multiple sectors to be entered into BADBLK.SYS even though there is only one defect. Defects in "hidden" areas need not be put in BADBLK.SYS since they cannot be read.

Documentation

This document is in printable and viewable form as XCERT.DOC

XCERT Hash Totals

XCERT.LIT	1.1(115)	254-372-276-076

The following is a list describing the steps XCERT follows:

1) Check if the drive is ok to certify
   A) Is the drive in the device table ?
   B) Is the drive mounted ?
   C) Is it the fist logical of the physical drive ?
2) Process switches
3) Ask user what type of drive
4) Get the drive serial number from the user
5) Get the list of defects from the user (cylinder, head, bytes from index)
6) Initialize the controller for this type of drive
7) If no switch present to not format the drive, then format the drive
8) Convert the cylinder, head, and byte into blocks
9) Do any extra checking for defects if required by drive
10) Sort the list of defective blocks eliminating duplicates
11) If we formatted then initialize the drive
   A) Write out the diagnostic cylinder
   B) Set up the MFD of the 1st logical
   C) Set up [1,2] on the 1st logical
   D) Write out BADBLK.SYS
   E) Write out BADBLK.LST if switch present
   F) Write the MFD and [1,2] for all other logicals

   ELSE

   A) Back up BADBLK.SYS to BADBLK.LST if it exists
   B) Write out BADBLK.SYS
   C) Write out BADBLK.LST if switch present
13) Print out the list of defects in head, cylinder, sector format
14) Exit XCERT

The only time a Winchester drive should be recertified is if, after the drive is already in use, the controller begins to report hard errors for the drive or frequent CRC errors for the same disk areas, indicating that new media flaws have occurred. In general, drives should only be recertified when areas of the drive become inaccessible.

We would like to encourage our dealers not to recertify Winchester disks unless it is absolutely necessary. Three major reasons *not* to recertify are:

1. The Winchester disks from Alpha Micro have already been fully certified and tested during system burn-in. Therefore, recertifying the disk is an unnecessary, time-consuming procedure.
2. In addition, sophisticated testing procedures used at Alpha Micro (which are not feasible for use in the field) allow us to make sure that the optimum BADBLK.SYS file is created for each drive. If the drive is recertified, it is not likely that this optimum BADBLK.SYS file will be matched, and the drive may be marginally less reliable in its handling of media flaws.
3. Finally, it is not necessary to recertify a drive in order to change the number of logical devices on that drive.

Changing the Number of Winchester Logical Devices without Recertifying

Although we have recommended in the past that a Winchester drive be recertified when changing the number of logical devices on that drive, we have developed a simple procedure for accomplishing this change that takes only a few minutes. To change the number of logical devices on an *empty* Winchester drive regardless of the number of logical devices with which the drive was originally recertified, follow the procedure below.

To change the number of logical devices on a drive that contains data, 1) make a backup of all the data on the drive, 2) check the backup media to make sure it is a valid copy of the data, 3) perform the procedure described below, and 4) copy the backup data to the drive.

NOTE: This procedure deletes all data from the Winchester disk. Therefore, if the Winchester drive you wish to change is the System Device, it is extremely important that you have a backup from which to boot the system. We would like to encourage you whenever possible to configure the Winchester drive as a pripheral to an existing system, and not as a System Device.

1. Use FIX420 to generate a new driver for the Winchester drive. Specify the number of logical devices for the drive to be used. Make a note of the bitmap size displayed by FIX420 for use later when modifying the system initialization command file.
2. Make a copy of the system initialization command file – SYSTEM.INI for AMOS systems and AMOSL.INI for AMOS/L systems – called TEST.INI.
3. Define the drive to the TEST file specifying the number of logical devices to be used. This means adding the correct DEVTBL and BITMAP commands for the drive, following the standard rules for defining a new Winchester drive to the system. Remember to name the devices using the three-character name assigned to the new driver created in Step 1 above.
4. If the Winchester disk is the System disk, use the MONGEN command to incorporate the new driver into a copy of the monitor. When MONGEN asks for the name of the new monitor, do no use the name of the normal monitor – use a name such as TEST.MON.
5. Physically connect the drive to the system.
6. MONTST with the TEST.INI file.
7. If the system does not boot normally, reboot with the system initialization command file and check the TEST.INI for mistakes.
8. If the system boots normally, boot under the regular system initialization file and make a bootable backup copy of PPNs 1,4, 1,6 and 2,2. Now use the SYSTAT command to check that the new Winchester logical devices are listed in the device section of the display. If the SYSTAT display appears to be correct – all jobs and devices are defined – continue with this procedure. If the display appears to be incorrect, press the RESET button to boot with the normal system initialization command file and view the TEST.INI to see what is wrong.

   Do not do anything that writes to the Winchester disk; writing to that disk while booted under the TEST.INI file will cause severe problems that can only be remedied by recertifying the disk.
9. MOUNT each of the Winchester logical devices.
10. Log into account [1,2] on the first logical device of the Winchester drive.
11. Use the LOAD command to load a copy of the BADBLK.SYS file in [1,2] into memory. This is a *very* important step – failure to do this will cause the BADBLK.SYS file to be lost in the later steps of this procedure and recertification will be necessary.
12. If the first logical device of the Winchester drive is not the System Disk, skip to step 14.
13. If the first logical device of the Winchester drive is the System Disk, log into account DSK0:[1,4] (the first logical device of the Winchester drive) and use the LOAD command to load the SYSACT and SAVE commands into memory. Programs that will be used to restore the System Disk after conversion is finished must also be loaded into memory. The next step will delete all files from DSK0:, destroying the System Disk. The loading of the above files will enable you to restore your System Disk.
14. Log into account [1,2] on the first logical device of the Winchester drive.
15. Use the SYSACT I command to initialize the first logical device of the Winchester drive.
16. Use the SAVE command to save the copy of the BADBLK.SYS file in memory to the disk: SAVE BADBLK.SYS

CAIR 3.4

FOR THE BIG 'L' SDS BOARDS

ALPHA	TDV	2	364-020-515-623		DVR:
BBBST	SBR	4	063-402-577-765	1.0	BAS:
BBINIT	CMD	1	615-511-072-076		CMD:
CAIR	SCP	27	114-254-161-744		SYS:
CAIR	SCR C	8			EACH USER PPN
CAIRLD	LIT	14	413-527-504-061	1.0	SYS:
CAIRPR	LIT	6	211-622-327-642	3.0(103)	SYS:
CAIRPS	ADR	1			SYS:
CAIRST	RUN	1	015-516-113-632		BAS:
TELVDO	TDV	2	163-700-254-112		DVR:
CAIR	LIT	25	672-035-176-124	3.4	[50,0]
CAIR	SYM	7	625-471-777-707		[50,0]

FOR AM1000 AND AM330 BOARDS

ALPHA	TDV	2	364-020-515-623		DVR:
BBBST	SBR	4	772-335-227-522	1.0	BAS:
BBINIT	CMD	1	615-511-072-076		CMD:
CAIR	SCP	27	114-254-161-744		SYS:
CAIR	SCR C	8			EACH USER PPN
CAIRLD	LIT	14	253-676-704-027	1.0	SYS:
CAIRPR	LIT	6	211-622-327-642	3.0(103)	SYS:
CAIRPS	ADR	1			SYS:
CAIRST	RUN	1	015-516-113-632		BAS:
TELVDO	TDV	2	163-700-254-112		DVR:
CAIR	LIT	25	357-722-446-147	3.4	[50,0]
CAIR	SYM	7	625-471-777-707		[50,0]

```
TRM.DVR[1,6]
STR.DVR[1,6]
VCR.DVR[1,6]
MSTAT.SYS[1,4]
MTU.DVR[1,6]
SCNWLD.SYS[1,4]
STRRES.LIT[1,4]
STRDIR.LIT[1,4]
STRSAV.LIT[1,4]
VCRRES.LIT[1,4]
VCRDIR.LIT[1,4]
VCRSAV.LIT[1,4]
TAPFIL.LIT[1,4]
FILTAP[1,4]
TAPDIR.LIT[1,4]
DIR.LIT[1,4]
MAP.LIT[1,4]
SYSTEM.LIT[1,4]
DSKANA.LIT[1,4]
SYSACT.LIT[1,4]
DSKDDT.LIT[1,4]
DUMP.LIT[1,4]
MOUNT.LIT[1,4]
COPY.LIT[1,4]
ERASE.LIT[1,4]
LOAD.LIT[1,4]
SAVE.LIT[1,4]
DEL.LIT[1,4]
LOG.LIT[1,4]
SYSTAT.LIT[1,4]
BADBLK.SYS[1,2]
BADBLK.LIT[1,4]
CRT415.LIT[1,4]
CRT420.LIT[1,4]
AMOSL.INI[1,4]
AMOSL.MON[1,4]
LSYS.MON[1,4]
PRINT1.INI[1,4]
DISK.DVR[1,6] YOUR DISK NAME
REDALL.LIT[1,4]
SET.LIT[1,4]
FIX420.LIT[1,4]
SYSTEM.PDF[7,0]
UNIQUE.SOV[1,20]
PRTSPC.DAT[7,6]
```

NAME

DETACH - detach terminal from job

USAGE

DETACH name

where "name" is the TRMDEF name in SYSTEM.INI

DESCRIPTION

DETACH disconnects the specified terminal from whatever job it is connected to.

COMMENTS

Detaching a terminal that is currently running is probably not very wise.

This program is reusable, meaning it can be loaded into a user's area and used over and over.

This program is reentrant, meaning it can be loaded into the system memory area and used over and over by any number of jobs at once.

PROGRAMMER

Larry White

BUGS

This program allows you to detach all terminals on the system.

DSKDDT HELP

FORMAT:

DSKDDT Devn:Block Devn = DEVICE (Disk), Block = BLOCK # you want.
DSKDDT DSK1:1 Would be DSK1:, block # 1. (DISK 1's MFD)

SUMMERY:

nnn/	Where nnn is the disk location you want to examine. This is the relative position of the location from the front of the block. (For example, 6/ displays the contents of the sixth and seventh bytes in the block).
nnn/NNN	Where nnn is the relative disk location you want to examine (enter in octal), and NNN is the octal data (two bytes) in which you want to replace the contents of nnn.
LINE-FEED	Dispalys the next two bytss of data in the block.
^	Displays the previous bytes of data in the block.
/	Displays location zero in the block (that is the very first two bytes in the block).
RETURN	No operation.
RUBOUT	Cancel current command line and displays XXX followed by TAB.
E	Rewrite the modified block and exit.
nN	Causes the current memory contents to be written to block "n" when an E command is given. This allows copying of one block to another.
O	Sets the entire block to minus one (177777 Octal or FFFF Hex, all ones). The data on the disk will not be changed until an E command is given.
Z	Set the entire block of data to zeros. The data on the disk will not be changed until an E command is given.
^C	- Exit without updating the block.

NOTE IN THE MFD:
Address 000772/ is the next link block in the MFD.
Address 000774/ is the preceding MFD link block.

NOTE IN THE UFD:
Address 000000/ or / is the link block pointer, just enter a /

LISTING FOR ERRORS OCCURRING IN BASIC PROGRAMS

1	Control-C interrupt
2	System error
3	Out of Memory
4	OUt of data
5	NEXT without FOR
6	RETURN without GOSUB
7	RESUME without ERROR
8	Subscript out of range
9	Floating point overflow
10	Divide by zero
11	Illegal function value
12	XCALL subroutine not found
13	File already open
14	IO to unopened file
15	Record size overflow
16	File specification error
17	File not found
18	Device not ready
19	Device full
20	Device error
21	Device in use
22	Illegal user code
23	Protection violation
24	Write protected
25	File type mismatch
26	Device does not exist
27	Bitmap kakput
28	Disk not mounted
29	File already exists
30	Redimensioned array
31	Illegal record number
32	Invalid filename
33	Stack overflow

.DUMP MFD

Master File Directory Dump of DSK0:

Block number 1, previous block link is 7, next block link is 7

```
000000    [1,2]          000006
000010    [1,4]          004136
000020    [1,6]          004142
000030    [2,2]          004144
000040    [7,0]          004145
000050    [7,1]          004146
000060    [7,6]          004147
000070    [7,7]          000042
000100    [7,11]         004150
000110    [10,1]         004153
000120    [10,2]         004154
000130    [25,25]        004205
000140    [0,0]          000000
000150    [0,0]          000000
000160    [0,0]          000000
```

The top line tells you that it is a DUMP of DSK0: Master File Directory.

The second line tells you the first block number of this MFD and the next link block if any.

The left most set of numbers is the starting address in the MFD in which you can work on a PPN (Numbers, Passwords, or UFD link blocks).

The center set of numbers are the PPN's

The right set of numbers are the link block to that PPN's User File Directory.

****

Below the left most set of numbers are the address in the MFD block 1.

The center set of numbers is the PPN number, UFD block link number, and the password of that account if any.

The PPN, Link Block Numbers are in Octal and the Passwords are packed in RAD50

.DSKDDT DSK0:1

```
/   402          [1,2] OPR:
2/   6           UFD LINK BLOCK NUMBER IS 6
4/   62073       FIRST 3 CHAR. OF PASSWORD (PACKED IN RAD50)
6/   73300       LAST 3 CHAR. OF PASSWORD (PACKED IN RAD50)
10/   404        [1,4] SYS:
12/   4136       UFD LINK BLOCK NUMBER IS 4136
14/   0          NO PASSWORD
16/   0          NO PASSWORD
20/   406        [1,6] DVR:
22/   4142       UFD LINK BLOCK NUMBER IS 4142
24/   0          NO PASSWORD
26/   0          NO PASSWORD
30/   1002       [2,2] CMD:
32/   4144       UFD LINK BLOCK NUMBER IS 4144
34/   12252      PASSWORD (PACKED IN RAD50) PACKED
```

```
36/   0          NOT USED ONLY 3 CHAR. PASSWORD
40/   3400       [7,0] LIB:
42/   4145       UFD LINK BLOCK NUMBER IS 4145
44/   0          NO PASSWORD
46/   0          NO PASSWORD
50/   3401       [7,1] HLP:
52/   4146       UFD LINK BLOCK NUMBER IS 4146
54/   0          NO PASSWORD
56/   0          NO PASSWORD
```

RAD50 PACKING/UNPACKING

RAD50 Packing Algorithm:

The packing algorithm for a 3-charactor input to a 16-bit RAD50 result is:

1. The first charactor code is multiplied by 3100 octal (50 X 50).
2. The second charactor code is multiplied by 50 octal and added to the first.
3. The third charactor code is added to the above to form the results.

JAB = 37252 J = 12, A = 1, B = 2

```
   3100      50                    12 X 3100 = 37200
 X   12    X 1                      1 X 50   =    50
--------    --                      2 X 1    =     2 +
   6200     50       2             ------------------
  3100                                        37252
---------
  37200
```

NOTE: THIS IS ALL DONE IN OCTAL (BASE 8).

RAD50 Unpacking Algorithm:

The unpacking algorithm merely reverses the above sequence to get the triplet.

1. The packed code is devided by 3100 octal, to get the first char. code.
2. The remainder packed code is devided by 50 octal, to get the second charactor code.
3. The third packed code is the left over of the above to form the results the third charactor code.

37252 = JAB 12 = J, 1 = A, 2 = B

```
          12                1                  2
        -------           -----
3100 | 37252        50 | 52
       3100              50
       -----             --
       06252              2
        6200
        ----
        0052
```

NOTE: THIS IS ALL DONE IN OCTAL (BASE 8).

RAD50 PACKING

CAHR.	#
BLANK	0
A	1
B	2
C	3
D	4
E	5
F	6
G	7
H	10
I	11
J	12
K	13
L	14

M	15
N	16
O	17
P	20
Q	21
R	22
S	23
T	24
U	25
V	26
W	27
X	30
Y	31
Z	32
&	33
.	34
0	36
1	37
2	40
3	41
4	42
5	43
6	44
7	45
8	46
9	47

*** SuperVUE ***

SuperVUE is used for word processing on the Alpha Micro. SuperVUE operates in two basic modes, DISPLAY MODE and COMMAND MODE. In Display Mode the text of the document being edited appears on the screen, and you may interactively examine and modify it. Command Mode is used for non editing operations, like disk i/o.

To invoke SuperVUE use the form:

.SV <filename>

where <filename> is the name of the document you wish to edit.

DISPLAY MODE

When SuperVUE is in Display Mode, it operates just like a typewriter. The CRT cursor corresponds to the carriage position on a typewriter. Since the entire document will not fit on the screen at one time, only a limited "window" is visible at any one time. This window may slide in any direction. You perform editing operations by pressing control keys.

DISPLAY MODE QUICK REFERENCE

right.....^L	insert char.....^F	next page.....^T
left.....^H	delete char.....^D	last page.....^R
up.....^K	delete word.....^V	home page.....^^
down.....^J	insert line.....^B	end page.....^E
next word.....^W	delete line.....^Z	next match.....^X
last word.....^A	join lines.....^O	reformat.....^S^F
column 1.....^U	erase line.....^RUB	center.....^S^C
end of line.....^N	erase eol.....^Y	auto-insert.....^Q
mark block.....^P	page break...^S RET	divide...../
clear block.....^P	math mode.....^S=	display.....=
move block.....^S^G	add.....+	enter.....@
copy block.....^S^D	subtract.....-	total.....T
erase block....^S^Z	multiply.....*	define macro...^S^X

COMMAND MODE QUICK REFERENCE

P)rint.....print document	H)elp.....system info
F)inish....save & return to AMOS	D)ir.....disk directory
Save.....save document	T)ype.....display a document
Q)uit.....abandon document	Amos.....temp return to AMOS
Format.....format document	S)earch....locate string
Sheets.....single sheet print	R)eplace...replace string
Y)ank.....merge document	Global.....replace w/o query
U)nyank....write block to disk	Page.....paginate document
SV.....edit another document	SVQ.....quit and edit document

DOT COMMANDS

page length.....LENGTH main title.....TITLE

top margin.....TOPMARGIN
title margin.....TITLEMARGIN
bottom margin.....BOTTOMMARGIN
left margin.....MARGIN

subtitle.....SUBTITLE
numeric title.....NUMTITLE
set page number.....NUMBER
set chapter number..CHAPTER

.DUMP UFD

Directory dump of block 4205, next block link is 0

Addr	Filename		Size	Active	Link	DSK0:
000002	BIGI	HLP	2	000326	004246	
000016	DSKDDT	HLP	4	000067	004250	
000032	JAB	LIT	3	000776	004213	
000046	JAB	M68	41	000776	004262	
000062	LOG	LIT	4	000154	004460	
000076	LOG	M68	41	000752	000011	
000112	:8ODSK	DDT	2	000230	004233	
000126	MFD	HLP	4	000730	004230	
000142	START	CMD	1	000035	000023	
000156	TEST	BAK	1	000474	004210	
000172	TEST	BAS	1	000543	000010	
000206	UFD	HLP	1	000002	004231	
000222	:8O	TMP	1	000002	004232	
000236			0	000000	000000	
000252			0	000000	000000	
000266			0	000000	000000	

The first line tells you the block number you dumped and next block link #.

The left most set of numbers are the address of the UFD block.

The next set is the FILENAME and EXTENSION (Unpacked in RAD50).

The center set is the total number of blocks in the file (Linked together).

The next set is how many active bytes there are in the last block of the file.

The last set are the block numbers of the first block in the file.
****

Below the left most set of numbers is the address bytes in the block of the UF

The right set is the file NAMES, EXT, SIZE of file, ACTIVE bytes in the last block of the file, and the first block in the file (LINK BLOCK).

All of the file NAMES and EXTENSIONS are packed in RAD50.

The very first byte is the BLOCK LINK pointing to the next UFD if there is on
.DSKDDT DSK0:4205

```
/   0           THE NEXT UFD BLOCK LINK, IF THERE IS ONE.
2/   6757       THE FIRST 3 CHAR. IN THE FILE NAME "BIG" (PACKED RAD50).
4/   34100      THE LAST 3 CHAR. IN FILE NAME "I" (PACKED RAD50).
6/   31760      THE FILE EXTENSION NAME "HLP" (PACKED RAD50).
10/   2         FILE SIZE (NUMBER OF BLOCKS IN THE FILE).
12/   326       THE NUMBER OF ACTIVE BYTES IN THE LAST BLOCK OF THE FILE.
14/   4246      THIS POINTS TO THE FIRST BLOCK OF THE FILE.
16/   16003     FIRST PART OF THE FILE NAME "DSK" (PACKED RAD50).
20/   14664     SECOND PART OF THE FILE NAME "DDT" (PACKED RAD50).
22/   6263      THE FILE EXTENSION "HLP" (PACKED RAD50)
24/   4         FILE SIZE (THIS FILE IS 4 BLOCKS LONG).
26/   67        THERE ARE 67 ACTIVE BYTES IN THE LAST BLOCK OF (DSKDDT.HLP).
30/   4250      THE FILE DSKDDT.HLP STARTS WITH BLOCK 4250 (OCTAL).
32/   37252
34/   0
```

```
36/   46174
40/   0
42/   776
44/   4213
```

Extended Tab Functions

0	Clear Screen / Normal Intensity
1	Cursor home (upper left corner)
2	Cursor return to column one
3	Cursor up one row
4	Cursor down one row
5	Cursor left one column
6	Cursor right one column
7	Lock keyboard
8	Unlock keyboard
9	Erase to end of line
10	Erase to end of screen
11	Enter half intensity
12	Enter normal intensity
13	Enable protected fields
14	Disable protected fields
15	Delete line
16	Insert line
17	Delete character
18	Insert character
19	Read cursor address
20	Graphics on
21	Graphics off
22	Ring bell
23	Display status line
24	Display user line
25	Print screen
26	Copy print on
27	Copy print off
28	Set normal screen
29	Normal/blank
30	Normal/blink
31	Set reverse screen
32	Reverse/blank
33	Reverse/blink
34	Normal/underline
35	Normal/underline/blank
36	Normal/underline/blink
37	Reverse/underline
38	Reverse/underline/blank
39	Reverse/underline/blink
40	Lock line
41	Unlock line
42	Clear unprotected to null

FACTORS WHICH INCREASE THE MONITOR SIZE
(For a 100T based system)

1.	"JOBS"	300 bytes	Per job, # of job comand lines has no effect.
2.	"TRMDEF"	46 bytes	Basic statement
	AM100T	208 bytes	First statement, second one adds 16 bytes.
	AM300	358 bytes	First statement, second one adds 16 bytes.
	TV950	484 bytes	First statement, second one adds 16 bytes.

example:
```
TRMDEF TRM1,AM300=1:16,TV950,60,60,15
TRMDEF TRM2,AM300=2:16,TV950,60,60,15
```

First line:
```
basic statement.............46 Bytes
Interface driver + 16 ....358 Bytes
Terminal driver + 16 .....484 Bytes
In width size ..............60 Bytes
In buffer size .............60 Bytes
Out buffer size (x 2) .... 30 Bytes
                           -----------
                           1038 Bytes
```

Second line:
```
Basic statement ............46 Bytes
Interface driver ...........16 Bytes
Terminal driver ............16 Bytes
In width ...................60 Bytes
In buffer ..................60 Bytes
Out buffer (x 2) ...........30 Bytes
                            ----------
                            228 Bytes
```

The basic Pseudo trmdef line is 46 bytes plus the buffer sizes.

3.	"MEMDEF"	18 Bytes	For the first one, all others, 12 bytes
4.	"MEMERR"	0 Bytes	
5.	"SYSMEM"	10 Bytes	Per line
6.	"DEVTBL"	16 Bytes	For each logical unit listed. Including "TRM", "MEM", & "RES"
7.	"BITMAP"	34 Bytes	Per line plus 2 X the bitmap size (in words)
	"BITMAP/S"	10 Bytes	????? not too sure about this one.
8.	"QUEUE"	16 Bytes	Per block requested
9.	"CLKFRQ"	0 Bytes	
10.	"SYSTEM"	12 Bytes	Plus the size of the program to be loaded.

FIX420.LIT "DRIVER PATCH"

```
COPY    PATCH

.=602           ; FOR 1.1A AND 1.2 ;.=576 FOR 1.1; .=600 FOR 1.0 AND BEFORE
WORD    22

END
```

"CRT420 PATCH"

```
COPY    PATCH

.=7612          ; FOR1.1A AND 1.2
WORD    22

END
```

HARDWARE CHANGES

ON THE AM1000 XEBEC BD. REMOVE RESISTOR PACK (CHIP) BY J5 (50 PIN CONNECTOR).
IF IT IS SOCKETED JUST REMOVE IT.
IF SOLDER TYPE REMOVE CHIP & INSTALL A 220 OHM RESISTOR BETWEEN PINS 1 AND 14.

4.2.1 BADBLK.SYS/CRT420 Important Notice

CRT420 must not be used to recertify a drive unless you have a copy of the original BADBLK.SYS file as shipped from Alpha Microsystems. This includes the 8" 32 megabyte Priam drive, both the 8 and 14" 60 megabyte drives and the 30 and 55 megabyte drives that come with the AM-1000 series systems.

The original BADBLK.SYS file contains information that cannot be duplicated in the field. If this original file is lost or corrupted, your only alternative is to send the drive to Alpha Micro for factory recertification.

If you have a copy of the original BADBLK.SYS or a listing of its contents, then the integrity of the drive may be restored while in the field.

Save a Copy of BADBLK.SYS

A copy of the original BADBLK.SYS must be made and kept accessible somewhere other than the system itself. A hard copy should be kept that lists all of the bad blocks contained in the BADBLK.SYS file or a copy of the BADBLK.SYS file itself should be stored on some external media such as VCR tape.

Only if you have a good copy of the original BADBLK.SYS file, can CRT420 be used on these drives without harm. If it is used, the original file must be restored to the disk prior to reloading the rest of the software.

WARNING: Be prepared to Restore from Backup

Anytime BADBLK.SYS is changed, the device needs to be re-MOUNTed to load the new information into the Alternate Track table and the rest of the software will then have to be cleared and restored from backup. Therefore, before any work is done on BADBLK.SYS have a current back-up of all software on the whole drive (not just the logical device that contains BADBLK.SYS). This is true if you are restoring the BADBLK.SYS file from a copy or if you modify the existing file using the BADBLK.SYS program.

Recovering BADBLK.SYS

Recovering BADBLK.SYS can be done by using the BADBLK program to add in all of the original bad blocks (or head and tracks) or by reinstalling the original BADBLK.SYS file.

Using the BADBLK Program

With a listing of the contents of the original BADBLK.SYS, you can use the BADBLK program to add in the blocks (or head and tracks) that were not picked up by CRT420. Documentation on how to use BADBLK can be found in the System Commands Reference Manual (DSS-10004-00 Rev. A05 for AMOS/L 1.2 or DWM-00100-49 Rev. A05 for AMOS 5.0).

This feature is available with AMOS/L Version 1.1a and later or AMOS Version 5.0 and later.

Restoring BADBLK.SYS from Backup

The technique of restoring BADBLK.SYS from a copy of the file is very similar regardless of where the copy of the file exists. The secret is to restore the BADBLK.SYS file onto the disk using a different name. The location of BADBLK.SYS is also important (it must reside within track 0).

If the wrong BADBLK.SYS exists on the disk and you need to restore the good one from tape, do the following:

1. LOG OPR:
2. RENAME BADCPY.SYS=BADBLK.SYS
3. VCRRES BADCPY.SYS=BADBLK.SYS
4. RENAME BADBLK.SYS=BADCPY.SYS

If there is no BADBLK.SYS on the device, the best approach is to SYSACT and initialize the disk. Then create a BADBLK.SYS using the BADBLK program, and perform the above steps.

A more detailed discussion on this issue may be found in the July, 1984 AMTS Journal, Vol. 6, No.7, 1.1.16 "Using CRT420 and BADBLK.SYS" or Software Volume, Section I, General Information.

If you have any additional questions, contact the Technical Support Group at (714) 641-5145.

LINEED

Version 1.5A for AMOS/L
Version 1.2 for AMOS

Input Line Editor
Installation Manual

by

Programming Sciences, Inc.
4651 Park Boulevard
San Diego, CA 92116
(619) 692-9280

Table of Contents

1.0 Description

The Programming Sciences line editor is a program which allows the editing of all terminal input throughout the Alpha Micro computer with VUE-like commands. LINEED immediately adds the following features to the system:

All AMOS level commands can be edited.

All BASIC, PASCAL, assembly, or any other language programs will allow full editing of terminal input without modification to any source code if they use the normal input statement.

2.0 Installation

2.1 Requirements

The requirements for LINEED are:

1. The program LINEED.PRG for AM-100 or AM-100/T systems or LINEED.LIT for AM-1000 or AM-100/L systems on a media compatible with the system. LINEED is available on single density single sided 8 inch floppy disk, VCR tape, or other media on special request.

2. Approximately 1.5K of system memory.

2.2 Initialization

Installation of LINEED requires several steps. Below is a list of the steps in the order that they should be taken. If a problem is encountered, see the section on troubleshooting or call your dealer or us at the number listed above.

1. Copy the Line Editor program, LINEED, from the received media onto the system in account DSK0:[1,4].

2. Copy SYSTEM.INI or AMOSL.INI into a test file (ie. TEST.INI) adding a command to include LINEED in system memory, and another invoking LINEED. See the sample AMOSL.INI or SYSTEM.INI below for an example.

3. Use MONTST to initialize the system using the new

initialization file (TEST.INI). This is to insure the system will still boot normally if there is a problem.

4. Verify that the system works as normal except that the editing features are active. If not, see the section below on troubleshooting.

5. Rename the tested initialization file (TEST.INI) into your normal initialization file (ie. SYSTEM.INI or AMOSL.INI)

6. All terminals will be in editing mode after LINEED is invoked.

2.2.1 Sample AMOSL.INI

Below is an example of an AMOSL.INI which includes the required statements to enable LINEED.

```
:T
Jobs 2
Jobalc Job1,Job2
Trmdef Trm1,AM1000=1:19200,Soroc,80,80,160
Trmdef Trm2,Am1000=2:9600,Soroc,80,80,160
Parity
Devtbl Trm,Res,Mem,/Vcr
Bitmap Dsk,1216,0
Queue 20
System Lineed   ; put LINEED.LIT in system memory          <<<<ADD
System
Lineed          ; turn on editing mode for all terminals   <<<<ADD
                ; MUST BE BEFORE ANY MEMORY ALLOCATION COMMANDS
Attach Trm2,Job2
Memory 0
```

2.2.2 Sample SYSTEM.INI

Below is an example of a SYSTEM.INI which includes the required statements to enable LINEED.

```
:T
Jobs Job1,Job2
Trmdef Trm1,Am300=1,Soroc,80,80,160
Trmdef Trm2,Am300=2,Soroc,80,80,160
Devtbl Dsk1,Trm,Res,Mem,/Vcr
Bitmap Dsk,606,0,1
Queue 20
System Lineed  ; put LINEED.PRG in system memory          <<<<ADD
System
Lineed         ; turn on editing mode for all terminals   <<<<ADD
Attach Trm2,Job2
Memory 0
```

2.3 Troubleshooting

LINEED is designed to be easy to install. If a problem is encountered, use the check list below.

1. Do you have a current copy of LINEED in DSK0:[1,4]?

2. Is LINEED in system memory?

3. Has it been invoked at least once?

4. Was it invoked before any memory allocation commands?

5. Do you have it enabled with Control-\?

If you have any problems with LINEED, call your dealer or us at the number listed above.

3.0 Operation

LINEED only allows editing when a program or the operating system is waiting for input (input wait state). At all other times, LINEED simply passes all input to AMOS which processes terminal input just as before LINEED was installed.

3.1 Data Entry

LINEED allows the typing of characters as usual with the addition of special editing commands. All characters and most editing commands can be typed faster than the screen is updated.

As characters are typed, they will overwrite existing characters unless

insert character mode (see Control Q) is active, in which case all characters to the right of the cursor will be shifted right as characters are typed. When in insert mode, if characters are typed faster than the input line can be updated on the screen, the character will be discarded.

3.2 Commands

Each editing command is described in detail below.

3.2.1 Cursor Left - Left Arrow or Control H

Moves the cursor to the left character-by-character until the beginning of the line or input field.

3.2.2 Cursor Right - Right Arrow or Control L

Moves the cursor to the right character-by-character until the end of the line or input field.

3.2.3 Backspace/Delete Previous Character - RUB or DEL

Deletes the previous character and shifts the remaining line left if insert mode is enabled (see Control-Q). If the cursor is at the beginning of the line or input field, then the character under the cursor is deleted. If the cursor is at the end of the line, the line will be shortened to reflect the deleted character.

3.2.4 Go to Beginning of Line - Control U

Moves the cursor to the beginning of the line or input field.

3.2.5 Go to End of Line - Control N

Moves the cursor to the last position on the line or input field.

3.2.6 Insert Character - Control F

Opens up space in the input line by shifting the remainder of the line or input field to the right by one position. If enough spaces are inserted so that the input line buffer is filled, the characters at the end of the line or input field will be lost.

3.2.7 Delete Character - Control D

Deletes the single character that the cursor is on and moves the remainder of the line or input field left one position.

3.2.8 Go to Next Word - Control W

Moves the cursor to the beginning of the next word. The cursor will not move past the end of the line or input field.

3.2.9 Go to Previous Word - Control A

Moves the cursor to the beginning of the previous or current word. The cursor will not move past the beginning of the line or input field.

3.2.10 Go to Character - Control G and Control X

Moves the cursor to the next occurence of the specified character. The character to search for is specified by typing control-G followed by the character. The character will be searched for without regard to upper or lower case. The searching sequence is in a right circular fashion, for example:

```
The quick brown fox
          ^          -- starting cursor position

^GQ                  -- control-G followed by Q

The quick brown fox
    ^                -- new cursor position
```

The search proceeded over "own fox", a match was not found, so the search continued at the beginning of the line.

By typing ^G^G (or the equivalent command ^X), the character to be searched for will be the character under the cursor. This command can be used to find the next match. For example,

```
Now is the time
  ^                     -- starting cursor position

^Gt                     -- control-G followed by T

Now is the time
       ^                -- new cursor position

^G^G                    -- control-G followed by control-G

Now is the time
            ^           -- next match found
```

Note: The Control-G and Control-X commands are only implemented on 68000 based systems.

3.2.11 Delete Word - Control V

Deletes everything from the cursor position to the beginning of the next word and moves the remainder of the input line left to fill the gap. Words are delimited by non-alphanumeric characters.

3.2.12 Erase from Cursor to End of Line - Control Y

Deletes everything from the cursor to the end of the line or input field.

3.2.13 Erase Entire Line - Control Z

Deletes the whole line or input field regardless of the cursor's position on the line or input field.

3.2.14 Swap Characters - Control S

Swaps the character under the cursor with the following character, leaving the cursor past the character pair.

Note: The Control-S command is only implemented on 68000 based systems.

3.2.15 Bring Back Previous Input String - Up Arrow or Control K

Replaces the current line or input field with the input before the last carriage return. The cursor will be positioned at the new end of line or input field. If the user is at AMOS command level, the previous input line will be placed on the next line.

3.2.16 Retype Current Line - Control R

Retypes the current line or input field without disturbing the position of the cursor. If the user is at AMOS command level, the current line will be retyped on the next line.

3.2.17 Enable/Disable Character Insert Mode - Control Q

Toggles character insert mode and beeps the terminal. When in insert mode, all characters to the right of the cursor will be shifted right as characters are typed, and the RUB key will shift the remainder of the line left. Insert mode will remain in effect for all lines of input until it is disabled.

3.2.18 Disable/Enable Editing - Control \

Toggles the input mode for the terminal between LINEED and AMOS and beeps the terminal. This command must be entered at the beginning of the line or input field. Control-\ does not effect other terminal's editing mode.

TAB characters are discarded in editing mode. If TABs are required in the input, editing mode must be disabled. See the section on Entering TABs below.

3.2.19 Carriage Return and Line Feed

Process terminal input line or field. Carriage return or Line Feed can be typed from any position in the field. Line Feeds are ignored at AMOS/L command level in the 68000 version of LINEED.

3.3 Command Summary

<-(Left Arrow)	Move Cursor Left
->(Right Arrow)	Move Cursor Right
RUB	Backspace/Delete Previous Character
^U	Go to Beginning of Line or Input Field
^N	Go to End of Line or Input Field
^F	Insert Character
^D	Delete Character
^W	Go to Next Word
^A	Go to Previous Word
*^G[char]	Go to character (right circular search)
*^G^G	Go to next occurence of character under cursor
*^X	Same as ^G^G
^V	Delete Word
^Y	Erase from Cursor to End of Line or Input Field
^Z	Erase Entire Line or Input Field
*^S	Swap Characters Under Cursor
Up Arrow	Bring Back Previous Line or Input Field
^R	Retype Current Line or Input Field
^Q	Toggle Insert Mode
^\	Enable/Disable editing (Toggle)

Note: The Caret (^) signifies that the control key is used.

* - 68000 based systems only

4.0 Non-Editing Modes

LINEED is temporarily disabled when one or more of the following conditions exists:

1. Control-\ is active.
2. Detached terminal.
3. Not in Input Wait state (program is running).
4. Command file is active (AM-100, AM-100/T only).
5. Image mode.
6. Data mode.
7. Echo off.
8. Over one line of input is buffered.
9. Control-S active.
10. Hard copy terminals.

5.0 Entering TABs

LINEED does not allow TABs in the input line or field. This is not a great disadvantage because TABs are rarely entered in AMOS command mode or as input to an applications or utility program. TABs can still be entered in the editor which is unaffected by LINEED.

If TABs are required, disable editing with Control-\. This command converts the terminal to the normal AMOS terminal input mode which does accept TABs.

LISTING FOR ERRORS OCCURRING IN BASIC PROGRAMS

1	Control-C interrupt
2	System error
3	Out of Memory
4	OUt of data
5	NEXT without FOR
6	RETURN without GOSUB
7	RESUME without ERROR
8	Subscript out of range
9	Floating point overflow
10	Divide by zero
11	Illegal function value
12	XCALL subroutine not found
13	File already open
14	IO to unopened file
15	Record size overflow
16	File specification error
17	File not found
18	Device not ready
19	Device full
20	Device error
21	Device in use
22	Illegal user code
23	Protection violation
24	Write protected
25	File type mismatch
26	Device does not exist
27	Bitmap kakput
28	Disk not mounted
29	File already exists
30	Redimensioned array
31	Illegal record number
32	Invalid filename
33	Stack overflow

4.3.1 XCERT and CRT420 Differences

Recently, AMSD has made available to our dealer base, XCERT, a maintenance and trouble shooting tool, primarily for our AM-1000 systems. We have received many questions concerning the differences between this program and CRT420, which is included with the operating system. We hope the following article will help to clarify these questions. There are two basic differences between the XCERT and CRT420. CRT420 assumes the disk drive is already formatted as an Alpha Micro drive; it also assumes the user knows where the defects are by block number, and is able to restore them to the defect map after completing the certification process. XCERT assumes the user knows where the defects are, by head, cylinder and byte as reported by the disk drive manufacturer and that the drive has never been formatted as an Alpha Micro drive.

CRT420

Basically, CRT420 will format the entire disk and create a BADBLK.SYS file. It will add all defective blocks it finds and reports to this file.

XCERT

XCERT, on the other hand, has several options that allows the user to restore or verify defects. They are:

/P Allows you to print a copy of the bad blocks from information entered.

/L Allows you to make a BADBLK.LST file of BADBLK.SYS from information entered.

/B Allows you to make a BADBLK.SYS file without formatting the drive from information entered.

Allows you to format the drive and make a BADBLK.SYS file from information entered.

These switches may be stacked. For example:

XCERT QTM0:/B/L

XCERT also checks all blocks in the remainder of the track, since this is not tested by the manufacturer, and all blocks that surround the known defective and found defective blocks.

Please note that XCERT requires the AMOS/L 1.2A operating system or later and is designed for use on 5 1/4" disk drives. The defect information XCERT needs to operate is recorded by the manufacturer of the disk drive and located on the HDA (Head Disk Assembly), of the disk drive.

For further information concerning this maintenance tool, please contact the Technical Services Group at (714) 957-8500 Ext. 479 or 390.