ADDENDUM to: SYSTEM SOFTWARE MANUAL, Revision 2.1

North Star Computers, Inc.
July 11, 1980

IMPORTANT: If your new Release 5.2 diskette will not boot up in your
computer, read the section of this addendum on the MOVER
program. You may need to use that program first.

This addendum describes the new features which have been added to
North Star System Software since Release 5.0, which is described in
Revision 2.1 of the SYSTEM SOFTWARE MANUAL.

SYSTEM SOFTWARE MANUAL ERRATA

Before proceeding, please make the following changes to the manual:

1. On page A-2 of the DOS section, file directory entry bytes
   10-11 are described as, "number of blocks in file", but
   should be described as, "number of sectors in file".

2. On page G-2 of the DOS section, one of the comments before
   DCOM reads, "ACC=NUMBER OF BLOCKS". It should read,
   "ACC=NUMBER OF SECTORS".

3. After a call to DCOM, the stack pointer will be left
   unchanged whether DCOM exitted via the return address on the
   stack or via the HDERR vector. If this information might
   ever be useful to you, please make a note of it on page G-2
   of the DOS section.

4. When performing DOS personalization, it is important to note
   that DOS 5.2DQ loads in two parts:

   Sectors 4-8 load into 100H-AFFH (assuming normal origin)
   Sectors 8-9 load into A00H-DFFH

   For example, if the personalization begins by typing "LF DOS
   4000", then an I/O subroutine destined to be loaded at
   2934H, or at 0A34H, will be found at 4834H, and a turnkey
   command string destined to be loaded at 2A55H, or at 0B55H,
   will be found at 4955H. Please make a note of this on page
   F-2 of the DOS section.

NEW FEATURES IN RELEASE 5.2

This section of this addendum describes the new features which
were added with Release 5.2. This information is presented
separately for the convenience of current users of the preceding
release of the software.

As before, there are two versions of Release 5.2. One version, 5.2S,
is for use with the older, single-density systems only. The other
version, 5.2DQ, is for use with double-density and quad systems,
only. The DOS's for these two systems are quite different
because they operate different disk hardware, and some of the
differences are noted below. But the remainder of the software is
independent of the type of system (single-density or
double-density/quad) on which it is used.

"MOVER" RELOCATION PROGRAM       *Siehe angehängtes Listing*

The MOVER program provides a simple method for relocating DOS, BASIC, and other programs, to special starting addresses. This procedure can be easily performed directly by the dealer or end user. (Special relocated versions of DOS and BASIC will no longer be available from North Star.)

In order to provide this program relocation capability, the Release 5.2 software diskette includes the MOVER program and a series of relocation key files. For each program on the diskette, which can be relocated, there is also a corresponding relocation key file. A relocation key file has type 3, and the file name is the name of the program to be relocated preceded by a hyphen, "-".

Programs on the software diskette are relocated by LOADing and RUNning the MOVER program under BASIC. When MOVER is run, it will make requests for relocation information on the console terminal. Then, it will sequentially attempt to relocate each program on the diskette for which a relocation key file exists.

If the relocation of a program on the diskette is not desired, the corresponding relocation key file should be deleted from the diskette before MOVER is run.

WARNING:   In order to save space, both in memory, and on the disk, this MOVER program writes the newly generated, relocated program into the same files from which the relocation key information is taken. This can only be done once. Therefore, it is very important that this only be done to diskette copies made from the Factory Master diskette, and never to the original Factory Master diskette, itself.

It may be necessary to use the MOVER program before any of the other new programs can be used at all. This will be the case in a system with a nonstandard bootstrap PROM set. This will also be the case in a double-density or quad system with no memory below 2000H, because the DOS, as supplied for these systems, loads at 100H. In these cases, MOVER should be run under an earlier release of North Star BASIC.

After the relocation is completed, and before the newly generated software can be used, it will be necessary to copy the new DOS into the first file position (sector 4) of a diskette, where it can be loaded by the bootstrap PROM. It might be appropriate to copy all of the relocated software to another diskette, starting with the DOS.

When using MOVER, it is possible to supply relocation information which leads to unreasonable or meaningless program relocations. In such cases, MOVER will cause either an OUT OF BOUNDS ERROR or the generation of unusable software (such as with DOS and BASIC overlapping). When this happens, do not be alarmed. Just recopy the diskette and repeat the process until you are completely satisfied with the results. Some of the mistakes which are expected to be the most common are described below.

The utlities are always followed in RAM by large areas (5 to 5.5 K) which are used for disk buffers. Like the programs, these areas may not wrap around the end of memory. Therefore, an attempt to locate the utilities above the space occupied by the standard disk controller (E800H-EBFFH) will fail.

As mentioned above, the MOVER works by comparing the relocation key files with the corresponding standard files on the same disk. Therefore, if one of the standard files has been personalized or otherwise altered, it will not relocate properly.

When an error occurs during the relocation of one of the files, the first part of the relocation key file will have been overwritten, but its type will still be 3. This file should be recopied from the master diskette right away. If instead MOVER is run again, it will attempt to use the overwritten portion of this file as relocation key data. This could produce very obscure results.

DOS

### I/O Device Status Routines

I/O device status check routines are now included in the user's I/O area to facilitate real time interactive and background operations. However, these routines are not used by any current software. Therefore, it is not necessary, at this time, for customized I/O systems to be regenerated with this feature included. The status routines are called with a device number in the accumulator and return the number of the device actually tested, if any, also in the accumulator. The input status check at 2041H or 141H, returns the Z flag true if the specified device has input data available. The output status check, at 2044H or 144H , returns the Z flag true if the specified device is ready to receive more data without delay. All other registers must be preserved by both routines. When these routines are not implemented, their positions in the jump table should contain the three byte sequence, AF 3D C9, which returns a "not ready" indication and an illegal device number (-1).

### Interrupts During Disk Transfers

Interrupts are now disabled during disk transfers and conditionally reenabled when permissable. Bit 7 of the RWCHK flag byte now controls this feature, while bit 0 of the same byte continues to control the read-after-write check. Interrupts will be left enabled after disk operations if and only if bit 7 has been set to one. This does not permit interrupts to serve as effectively as the OFTEN call for such things as type ahead, software clocks, or modem service because of the length of time that they must be disabled. But for extremely rare events, such as RAM parity errors, they may be appropriate.

Parity Errors
A routine is provided, when the DOS is personalized for the Horizon, which prints a message at the console if a memory failure is ever detected. To enable this feature, the RAM boards should be set to generate vectored interrupt five and the interrupt control flag, bit 7 of RWCHK, in the working copy of the DOS should be set to one. If the message, "RAM PARITY ERROR," appears on the terminal, you may press RETURN to continue with the parity check disarmed or press any other key to reboot (after putting the appropriate diskette into drive one).

DLOOK Error Return
The library routine, DLOOK, now returns a zero in the accumulator if an error was detected in the syntax of the specified filename. Otherwise, it returns the implied drive number, as it did in earlier releases.

Disk Controller Address
There is now a byte in the jump table at 203AH or 13AH, which indicates the origin of the disk controller. This information is provided for use by any program which may need to access the disk controller directly, such as to reboot the system by branching to the address of the bootstrap PROM.

DOS FOR DOUBLE-DENSITY AND QUAD

Speed Improvement
The low level reading and writing routines in the DOS have been modified for an increase in overall throughput. Some speed improvements will be noticed during copy operations and during extensive access to data files.

CK UTILITY

The CK Utility provides an easy way for the dealer or end user to verify that the contents of a Factory Master diskette are precisely correct, and have not been corrupted or accidentally changed, after leaving the factory. Running this utility computes an overall check code for an entire diskette. The number it displays can then be compared against the correct check value for any standard diskette, as given below. If the check value does not compare correctly, then the diskette should be replaced with a correct diskette.

The correct check values for the four standard diskettes are:

| Name | Part Number | Check Value |
|------|-------------|-------------|
| HORIZON-S | 11002C | 21516 |
| HORIZON-DQ | 11001B | 58777 |
| MDS-S | 11008C | 54115 |
| MDS-DQ | 11007B | 38179 |

## BASIC

### Longer Maximum Line Length and Suppression of Automatic Carriage Returns
The LINE statement can now be used to set the line length for any
device to as much as 165 characters. This statement will also now
accept an optional, additional argument, which must be a numeric
expression preceded by a comma. If this expression evaluates to
zero, BASIC will stop automatically sending carriage returns and
line feeds to the specified output device whenever its line length
is reached. If the expression evaluates to a non-zero value,
automatic generation of carriage returns and line feeds will be
resumed on the specified device. This feature may be useful when
sending something other than normal text to an output device.

### File Size Function
A new function, FILESIZE(N), returns the size in blocks of a
currently open file specified by the numeric expression, N.

### File Pointer Function
Similarly, another new function, FILEPTR(N), returns the current
position of the file pointer in currently open file, N. This
function may be useful in the numeric expression which specifies the
new position in a file for a random READ or WRITE statement.

### File Pointer Positioning
A WRITE statement of the following form:

    WRITE #N %P, NOENDMARK

will set the file pointer in any currently open file N, to any valid
position P, for a subsequent sequential READ or WRITE statement.
This statement may be used to reset a file pointer to zero without
closing and reopening the file.

## RAMTEST

Two copies of a new memory test program are provided, which occupy
different areas of memory so that all of memory can be tested. This
test is designed for use in Horizons only, requires a Z-80
processor, and does all of its I/O through the Horizon's standard
serial port.

RAMTEST always tests memory in 1K regions. It maintains a 64 x 8 matrix of memory error status. Whenever this matrix is typed out, the 64 characters per line represent the 64K address space, left to right, starting at zero. The line for bit 0 is typed first, the line for bit 7 last. The error matrix is cumulative. This means that once a bit is marked bad, it will not be marked good by subsequent passes.

Phase 0 is only performed once. It makes a quick judgment of the status of the memory system by testing only the first byte of each region. It then types the error matrix, with entries meaning:
      M = RAM memory
      - = no memory
      P = PROM or ROM
      * = region reserved for test program
On subsequent passes only the "M" regions are tested. Errors change the "M" to "?".

Phase 1 tests with a pattern of all zeroes.

Phase 2 tests with a pattern of all ones.

Phase 3 tests with a fixed pattern of 19 bytes repeated for the entire region. The test is done 38 times, with each byte of the pattern getting to be first, and incrementing memory both forward and backward.

Phase 4 tests with a pattern calculated to stress address buffers.

Phase 5 tests that a data pattern in each region survives stores to all other "M", "-", or "P" regions.

Phase 6 tests with an algorithm calculated to stress the ability of dynamic RAM cells to retain information in the face of noise in adjacent cells.

Phase "M" tests the ability of each region to execute programs. First a subroutine of 1023 NOP's and a RET is called several times. Next, a program calculated to create electrical noise is called, which moves itself about while performing calculatons. If the correct answer is returned, an "M" is typed. If the wrong answer is returned, a "?" is typed. If the subroutine fails to return, most likely nothing is typed, leaving the terminal's cursor indicating the offending region.

Phase "-" tests a memory board's ability to fetch an opcode, operand sequence from different memory chips, testing that the first chip deselects properly. This phase is performed only on even numbered passes to ensure that the computer first executes all other numbered phases.

Each full pass begins by typing out a pass counter which runs from "AA" to "ZZ" and then repeats. The following character is an exclamation point if any errors have occurred since the test began, or a space otherwise.

If a control C from the keyboard is detected (through the Horizon's standard serial port), the program jumps to location E800h to reboot. If any other character is detected, the error matrix is typed. Thus evidence of a single error is not lost, even if it has scrolled off of a CRT terminal.

The program assumes the presence of a North Star dual-density disk controller, which will generate wait states whenever location E918H is fetched. Obviously, the program will still run without wait states. A fetch of location E918H has no effect on a single-density controller. Wait states are used in phases 6 and "M".

For North Star RAMS, parity control is output on port C0H. Phase 0 attempts to light the error LED if any byte had bad parity on entering the program. After phase 0, the LED should light only on the occurrence of an error. A scan for correct parity is also done at the end of each phase. A chip location diagram for North Star RAM boards is included at the end of this addendum.

## Z80 Horizon I/O Routines

The I/O routines for Horizon computers supplied with Release 5.2 are written in Z80 code. This was done to make room for the parity message feature and to make it usable in systems which do not have RAM at zero. However, except for the RAMTEST, the remainder of the software on a Release 5.2 diskette is still entirely 8080 compatible.

## BUGS FIXED IN RELEASE 5.2

### DOS
The CR command now fails properly when given an illegal filename.

The read-after-write check option now works with the IN command.

Hard disk error messages are now routed to device 0.

### DOS for Single-Density
After "PRESS RETURN TO CONTINUE" appears in a directory list, device 0 is now specified in the call to CIN.

After RETURN is pressed to continue, a carriage return and a line feed are now echoed to the console.

The auto start option can now be set up even without using a fresh copy of the DOS.

When the auto start feature is used, the DOS herald is not displayed.

### DOS for Double-Density and Quad
The DOS will now boot up properly regardless of the previous contents of memory at its load address.

Directory listings will no longer cause the system to hang up for an undue length of time in the case of uninitialized or poorly mounted diskettes.

Directory listings also will not damage any software which may have overlaid the latter part of the DOS, starting at 2A00H.

OFTEN is now called during seeks on drives with fast track-to-track stepping.

### CO Utility
The proper way to compact overlapping files has never been agreed upon. Therefore, CO will no longer offer to proceed in the presence of conflicts.

CO will no longer offer to convert a purely double density diskette to double-density.

Density mismatch errors will now be handled properly in all cases.

### CF Utility
Small files can now be copied into very large files of 512 blocks or more.

Like the CR command, CF will now fail to create a new file given an illegal filename.

Like the CO utlity, CF will now handle density mismatches properly in all cases.

BASIC

The obscure bug involving the CREATE statement and its affect on several other, seemingly unrelated statements has now been fixed.

Numeric overflows will now be detected properly when dividing very large numbers by very small numbers, even without a Hardware Foating Point Board.

Use of user defined functions in the THEN clause of an IF statement will no longer prohibit the use of an ELSE clause in the same statement.

The DEF statement can now be followed by additional statements on the same line, specifically, the problem of following a DEF statement with one containing an equal sign, "=", has now been solved.

## NEW RELEASE 5.1 FEATURES

This section describes the new features incorporated with Release 5.1 system software.

The format of a double sided diskette is as follows:  Side A is  the same as a single sided diskette and can be used in single sided disk drives.  Side B holds tracks 35 through  69, with  track  35  at  the inside (opposite track 34),  and  with  track  69  at  the  outside (opposite track 0). The directory is still on side A, but entries in it may refer to disk addresses up through 699.  On a  double  sided, double-density diskette, a file which does not overlap the directory area may now be  as  large  as  1392  blocks  (384K  bytes,  or  696 sectors). Double sided diskettes may be used in single sided drives, but only the data on side A  may  be  accessed.  If  a  file  "wraps around" to side B, only the part of it on side A can be accessed in a single sided drive. Any attempt to access the remainder of such  a file will cause an error.

Single  density DOS 5.1S now supports the CD, CF, CO, and DT utility programs, and has all the features of the double-density DOS  except the  OFTEN  call  and,  of  course,  double-density  or double sided operation. In the single-density DOS, note that the  JMP  instruction at  2007H  or  0107H  has  nothing  to do with OFTEN and must not be altered.

The CD, CF, CO, and DT utlity programs  included  with  Release  5.1 system  software  are not the same as those that were supplied with Release 5.0 and will not work properly with that or any earlier DOS. These  utilities  will  now support double sided operations when used with the DOS 5.1DQ.

There is one new personalization byte, CONFG, at 2034H or  0134H  in standard  versions  of  Release  5.1 DOS. It tells the DOS and other system software two things about each  of  the  four  possible  disk drives in the system:

   1. The  upper  four  bits  tell  which drives are double sided. Bits 7 down through  4 correspond  to  drives 1  through  4, respectively.  Each bit must be 0 if the corresponding drive is single sided, or 1 to allow  double  sided  operation  of that drive.

   2. North  Star's  new,  double sided drives are capable of much faster stepping between  tracks.  The  lower  four  bits  of CONFG  tell  the  DOS which drives have this feature. Bits 0 through 3 correspond to drives 1  through  4,  respectively. Each  bit must be 0 if the corresponding drive can only step at normal speed, or 1 to take advantage of fast-stepping.

For example, a CONFG value of 5AH (01011010) indicates that only drives 2 and 4 are double sided, fast stepping drives. The proper CONFG value for four quad capacity drives is OFFH. DOS on factory master diskettes is supplied with zero in the CONFG byte, because any drive can be operated at normal speed, using side A only.

Note that the CONFG byte exists in both single density DOS 5.1S and dual density DOS 5.1DQ. The CONFG byte in the single density 5.1S must never be changed from its original zero value, because the single density system supports neither double sided diskette access nor fast-stepping.

The personalization process for DOS Release 5.1S is the same as that for Release 4 DOS, except that the jump table, after and including address 200DH, corresponds to the description of the Release 5.0 jump table in the SYSTEM SOFTWARE MANUAL. In particular, such features as the PAGES byte, the AUTOSTART flag, and a pointer to the command buffer, which have been in the double-density DOS jump table since release 5.0, are now contained within the jump table for the single-density DOS, 5.1S, as well.

The procedure for personalizing the dual density DOS remains the same as that given in the SYSTEM SOFTWARE MANUAL, except that the proper value for CONFG should be set in the new copy of DOS (in the workspace RAM) before it is saved on the personalized diskette.

Beginning with Release 5.1, DOS on factory master diskettes will have the read-after-write option enabled.

Two diskettes are provided with all standard HORIZON computer systems and MDS Micro Disk Systems: a FACTORY MASTER DISKETTE containing system software, and a blank to be used in preparing the WORKING DISKETTE as described in the SYSTEM SOFTWARE MANUAL. The blank included with quad-capacity systems is certified for full double sided operation, but the Factory Master system software diskette for ALL systems will remain single sided only, because recording upon the factory master, or removing its write protect tab, should never be done.

Standard Release 5.1 BASIC is now configured to use 24K of RAM, starting at address 2000H or 0100H. (This now includes over 8K for a user program and data.) Previous releases were configured for 16K of RAM.

Users of FPBASIC in single-density systems should note that the standard address of the floating point board was changed from DFFOH to EFFOH as of system software Release 5.0. At this time, a new personalization byte, FPBADDR (ORG + 21H), was added to FPBASIC (not regular BASIC). Changing FPBADDR enables FPBASIC to make use of a floating point board with address other that EFFOH. See DISCUSSION: PERSONALIZING BASIC in the SYSTEM SOFTWARE MANUAL for further details.

CHIP LOCATOR FOR NORTH STAR RAM 16:

| 0000 | 2000 | 4000 | 6000 | 8000 | A000 | C000 | E000 |
|------|------|------|------|------|------|------|------|
| 14D-14C | 14B-14A | 14D-14C | 14B-14A | 14D-14C | 14B-14A | 14D-14C | 14B-14A |
| 13D-13C | 13B-13A | 13D-13C | 13B-13A | 13D-13C | 13B-13A | 13D-13C | 13B-13A |
| 15D-15C | 15B-15A | 15D-15C | 15B-15A | 15D-15C | 15B-15A | 15D-15C | 15B-15A |
| 16D-16C | 16B-16A | 16D-16C | 16B-16A | 16D-16C | 16B-16A | 16D-16C | 16B-16A |
| 11D-11C | 11B-11A | 11D-11C | 11B-11A | 11D-11C | 11B-11A | 11D-11C | 11B-11A |
| 10D-10C | 10B-10A | 10D-10C | 10B-10A | 10D-10C | 10B-10A | 10D-10C | 10B-10A |
| 12D-12C | 12B-12A | 12D-12C | 12B-12A | 12D-12C | 12B-12A | 12D-12C | 12B-12A |
| 17D-17C | 17B-17A | 17D-17C | 17B-17A | 17D-17C | 17B-17A | 17D-17C | 17B-17A |
| | | | | | | | |
| -------- | MMMM**MM | -------- | 14B-14A | 14D-14C | -------- | -------- | --P----- |
| -------- | MMMM**MM | -------- | 13B-13A | 13D-13C | -------- | -------- | --P----- |
| -------- | MMMM**MM | -------- | 15B-15A | 15D-15C | -------- | -------- | --P----- |
| -------- | MMMM**MM | -------- | 16B-16A | 16D-16C | -------- | -------- | --P----- |
| -------- | MMMM**MM | -------- | 11B-11A | 11D-11C | -------- | -------- | --P----- |
| -------- | MMMM**MM | -------- | 10B-10A | 10D-10C | -------- | -------- | --P----- |
| -------- | MMMM**MM | -------- | 12B-12A | 12D-12C | -------- | -------- | --P----- |
| -------- | MMMM**MM | -------- | 17B-17A | 17D-17C | -------- | -------- | --P----- |

      ^ good board      ^ board under test

CHIP LOCATOR FOR NORTH STAR RAM 32:

| 0000 | 2000 | 4000 | 6000 | 8000 | A000 | C000 | E000 |
|------|------|------|------|------|------|------|------|
| --14D--- | --14C--- | --14B--- | --14A--- | --14D--- | --14C--- | --14B--- | --14A--- |
| --13D--- | --13C--- | --13B--- | --13A--- | --13D--- | --13C--- | --13B--- | --13A--- |
| --15D--- | --15C--- | --15B--- | --15A--- | --15D--- | --15C--- | --15B--- | --15A--- |
| --16D--- | --16C--- | --16B--- | --16A--- | --16D--- | --16C--- | --16B--- | --16A--- |
| --11D--- | --11C--- | --11B--- | --11A--- | --11D--- | --11C--- | --11B--- | --11A--- |
| --10D--- | --10C--- | --10B--- | --10A--- | --10D--- | --10C--- | --10B--- | --10A--- |
| --12D--- | --12C--- | --12B--- | --12A--- | --12D--- | --12C--- | --12B--- | --12A--- |
| --17D--- | --17C--- | --17B--- | --17A--- | --17D--- | --17C--- | --17B--- | --17A--- |
| | | | | | | | |
| -------- | MMMM**MM | -------- | --14A--- | --14D--- | --14C--- | --14B--- | --P----- |
| -------- | MMMM**MM | -------- | --13A--- | --13D--- | --13C--- | --13B--- | --P----- |
| -------- | MMMM**MM | -------- | --15A--- | --15D--- | --15C--- | --15B--- | --P----- |
| -------- | MMMM**MM | -------- | --16A--- | --16D--- | --16C--- | --16B--- | --P----- |
| -------- | MMMM**MM | -------- | --11A--- | --11D--- | --11C--- | --11B--- | --P----- |
| -------- | MMMM**MM | -------- | --10A--- | --10D--- | --10C--- | --10B--- | --P----- |
| -------- | MMMM**MM | -------- | --12A--- | --12D--- | --12C--- | --12B--- | --P----- |
| -------- | MMMM**MM | -------- | --17A--- | --17D--- | --17C--- | --17B--- | --P----- |

      ^ good board      ^ board under test

NOTES:
1. Each column of the RAMTEST table represents 1K of address space.
2. The columns are grouped by eights; each group corresponds to one of the selection switches on a North Star RAM board.
3. The first row printed is for bit 0, the last is for bit 7.
4. Meaning of designators:
   M = good memory
   ? = questionable memory
   P = PROM/ROM
   - = no memory