NSE

North Star Horizon Z80 Computer Emulator.

GUI Version

20th April 2021

This Page is left Blank

TABLE OF CONTENTS

NSE APPENDICES

IMAGES AND PRINTOUTS

1. INTRODUCTORY INFORMATION

1.1 Overview

NSE emulates the late 1970s to early 1980s North Star Horizon Z80 Computer.

NSE uses disk-image files which may contain any of North Star Computers' Disk Operating Systems of the period: North Star DOS (NSDOS), CP/M, UCSD Pascal, etc.

The original North Star Horizon in 1978 possessed a single-density floppy-disk-controller which used a single side of a 5-inch, 35-track, hard-sectored floppy disk with ten 256-byte sectors per track giving 88K of storage.

Later models used a double-density-controller which could access both sides of the disk and used 512-byte sectors giving 350K of storage. The double-density controller was not able to boot from a single-density disk but was able to read from and write on it.

Later again, hard-disk capability was added to the North Star Horizon.

NSE is constructed from two modules. The first module, the GXE Z80 Emulator Toolkit, contains the 64K of RAM, a Z80 microprocessor emulator, and a display screen. Drop-down menus emulate the operator's interaction with the hardware, such as inserting or removing floppy disks, and organizing the interaction between the host linux machine and the virtual Z80 machine. This first module is installed as a Graphical User Interface (GUI) and can be called by NSE or any other Z80-based emulator software, such as CPZ (ICMS CPZ-48000 emulator) or MD11 (Morrow Designs MD11 emulator).

The second module consists of the North Star Horizon-specific components, such as the data and control ports of the serial and parallel I/O, the single and double-density floppy-disk controllers with their boot PROMS, and the fixed-disk controllers

1.2 OTHER Z80 EMULATORS

A second Z80 CP/M emulator is included, called CPZ. This emulator is a virtual ICM CPZ-48000 single-board-computer. It uses disk-images which are virtual 8" disks, both single-density, single-sided and double-density, double-sided. This emulator also uses the GUI Template and is practically identical in usages and appearance as NSE.

A third Z80 CP/M emulator is in construction and is called MD11. This emulator is a virtual Morrow Designs system. It can be booted from an 8” disk-image, but is normally booted from the first of the four 32 megabyte hard drive images. Each of the four hard drive images contains four 8-megabyte drives. The sixteen 8-megabyte drives are allocated from A: to P:

1.3 ATTRIBUTIONS FOR OTHERS' CODE in NSE

NSE's Z80 emulation code pretty much comes from yaze, a CP/M emulator written by Frank Cringle. North Star specific amendments such as memory-mapped floppy-disk I/O, and a few other additions such as Mode 2 interrupt code were made by Jack Strangio.

NSE's Z80 disassembly code comes from Marat Fayzullin's 1999 DAsm code with some local alterations.

The rest of NSE cannot be blamed on anyone else but myself.
Jack Strangio, March 2020

1.4 THANKS

I have only the greatest appreciation for all those who have helped me in my rather idiosyncratic quest to write an emulator of the North Star Horizon. The Horizon was my first computer which took more than 40 hours to build during the course of several weeks in late 1978. The thousands of solder-joints literally burned-out a new soldering iron. It says a lot for the quality of the instruction manual that most of the time I really had no idea what each step did but at the end (once my half-dozen wiring mistakes were fixed) I had assembled a computer which worked perfectly.

I'd like to mention a few of the people who have generously helped me:

Dave Dunfield, who gave me a lot of help in many different areas. Often, just the fact that a disk-image worked on his HORIZON.COM emulator and not on my NSE emulator showed me that I had to find yet one more bug. He also had quite a few North Star floppy disk-image files.

The Late Don Maslin, who got me started on the double-density floppy work by transferring a lot of data from my old 10-sectored disks to disk-image files.

Martin Brown, who helped me along the way with scanning old Disk-Controller manuals, without which I was more clueless than usual.

Howard Harte , whose regard for old computers means he has taken the trouble to maintain lots of North Star Manuals:

http://www.hartetechnologies.com/manuals/Northstar/

Bitsavers.org. (http://www.bitsavers.org/bits/NorthStar/). Thanks to them, there are still quite a few disk-image files around for the North Star Horizon.

Allison Parent, for indicating where I could get hold of information regarding the HD5X controller board.

1.5 SEE ALSO ...

The horizon.com emulator for MSDOS by Dave Dunfield at Dave's Old Computers Website (http://www.classiccmp.org/dunfield/index.htm).

Dave also has lots of stuff regarding the North Star Horizon and other old 8-bit computers from the 70's and 80's.

1.6 FLOPPY DISKS AND A HARD DISK SUPPLIED WITH NSE

Several floppy disks are supplied with NSE to get you up and going quickly. They are stored in the 'disks' subdirectory. These archive disks have been renamed to allow their uses to be self-explanatory. The original names are also given here.

HDCPM01.NSI	(was D03B01.NSI)	North Star CP/M Boot Disk for Hard Disks
HDCPMA1.NSI	(altered D03B01.NSI)	CP/M disk which looks to A: for executables
HDOS22BOOT.NSI	(was D04B01.NSI)	North Star HDOS 2.2 Boot Disk
HDOS22REC.NSI	(was D07B01.NSI)	North Star HDOS 2.2 Inital Recovery Disk
NSDOS_51S.NSI	(Single-Sided/Single-Density Disk)	NOTE: Must be booted SS/SD (See page 25)

One hard-disk is also supplied as a sample. It is a CM10E-type hard disk storing 10 Megabytes. It has been pre-formatted and 'recovered'. On it are several CP/M virtual disks: CPMA, CPMB, CPMC, CPMD, CPME, CPMF, CPMG and CPMH. The first two floppy drives have been allocated to CP/M drives I: and J:

When CP/M is booted with the CP/M Boot Floppy Disk, it will take you to the A: directory on the hard-disk. This is counter-intuitive, as you would expect to have A: on the floppy. But that's how North Star worked it.

1.7 SCREEN VIEWS

(Note: Most of the screen images included in this User Guide do not render well. They look better when displayed dot-for-dot as screenshots. Find them in the 'screenshots' directory of the downloaded tarball.)

NSE is GTK+ based. When NSE starts up it will look at the screen-resolution and display a 'terminal' of a size that is suitable for that resolution. The 'large' terminal (110 chars wide, 45 lines) will fit on a 1920x1080 screen. The 'medium' terminal (96 chars wide, 36 lines) will fit on an intermediate resolution laptop. Then there is the 'small' terminal which is actually the historical standard-sized serial terminal of 80 chars wide and 25 lines high. Both the medium and small terminal displays can be specified with a command-line option.

(Fig 1, Page 8: Three NSE Start-Up Screens. (110x45, 96x36, 80x25 screen formats)

(Fig. 2, Page 8: CP/M Splash Screen with Directory Listing)

NSE looks like a typical "green-screen" terminal of the 70's-80's period, in particular it will default to be a terminal which acts very similar to Televideo 925/ Soroc 120/ ADM3A terminals.

(Fig 3, Page 9: WordStar running in high and wide screen format.)

(Fig 4, Page 10: HDOS running in high screen format)

NOTE: A more comprehensive list of screen-grabs shown throughout the manual is given on Page 4.

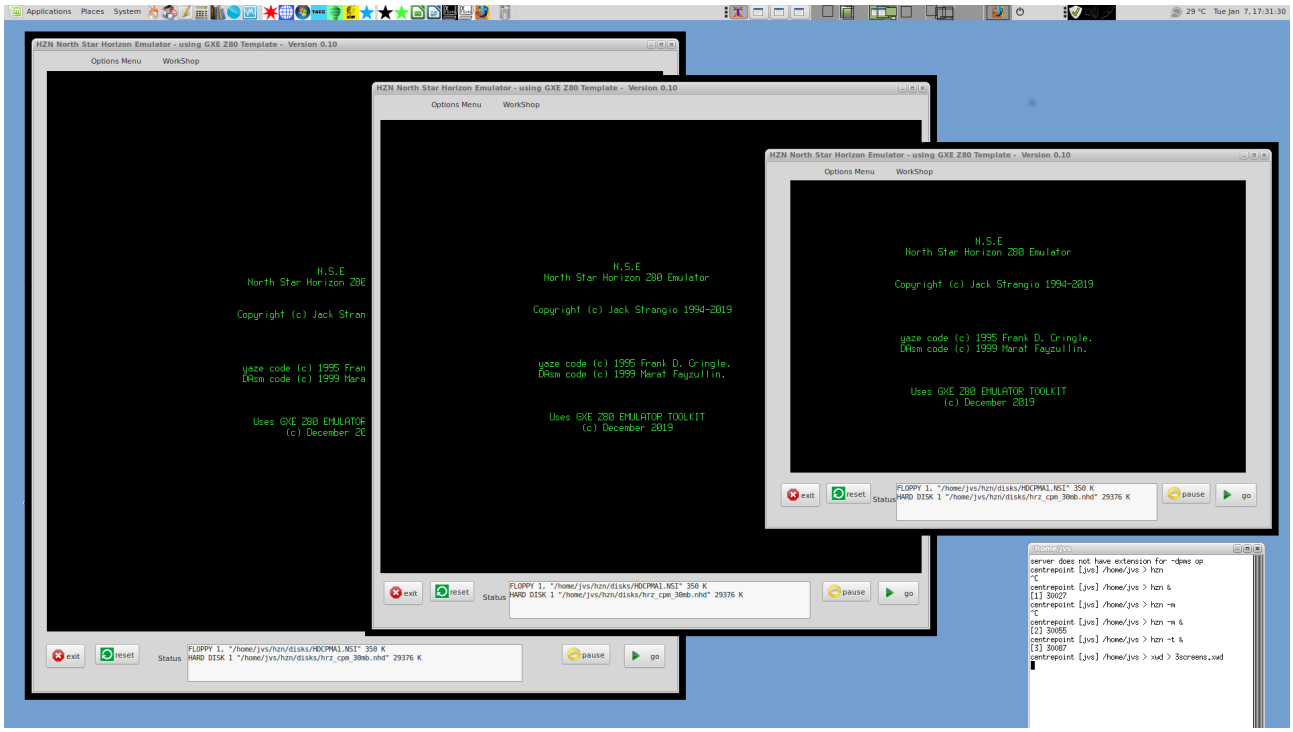

Fig 1. Three NSE startup screens, running on a 1920x1080 resolution screen in 110x45 display, 96x36 display, and in 80x25 character screen formats.

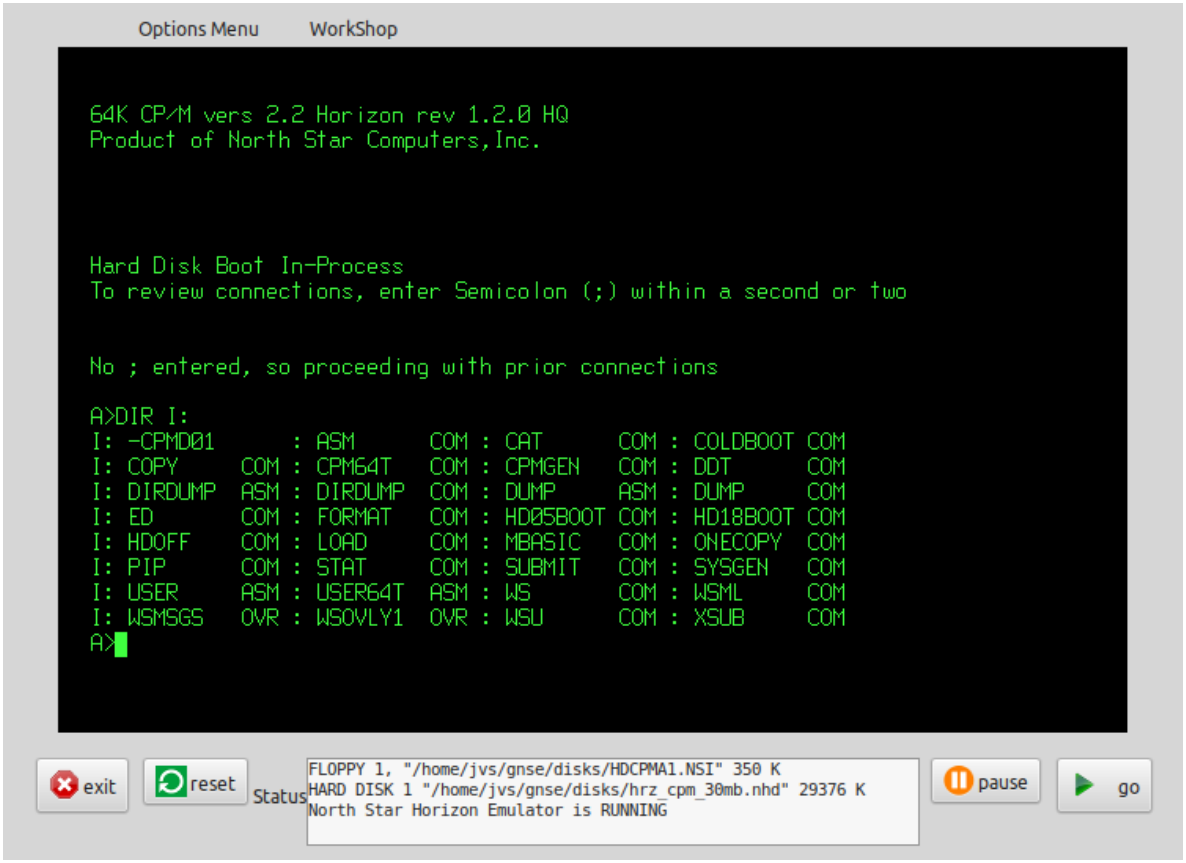

Fig 2. NSE splash screen followed by CP/M directory listing.

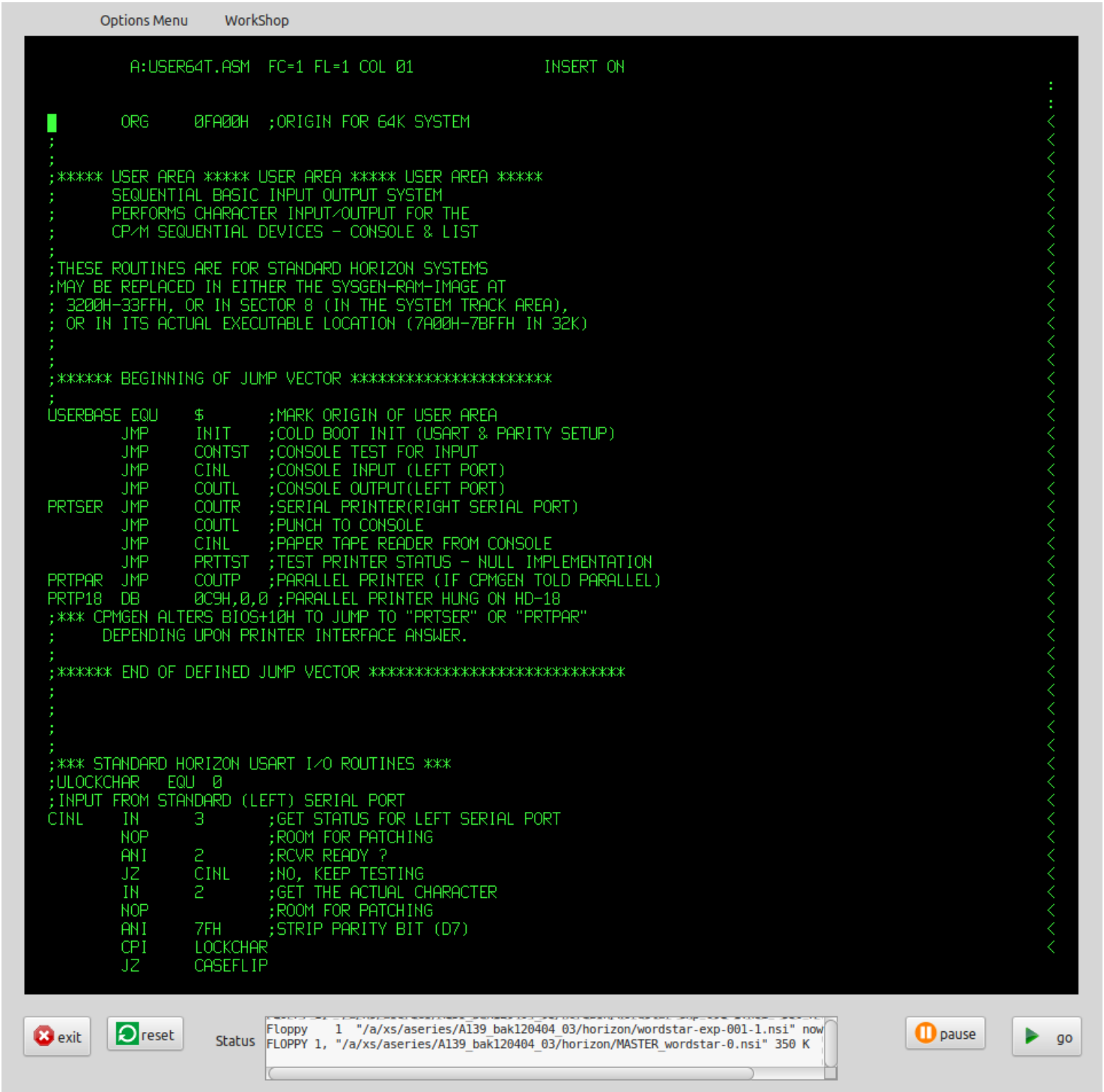

Fig 3. A custom-configured version of Word Star running in a high and wide screen format (110 chars,45 lines)

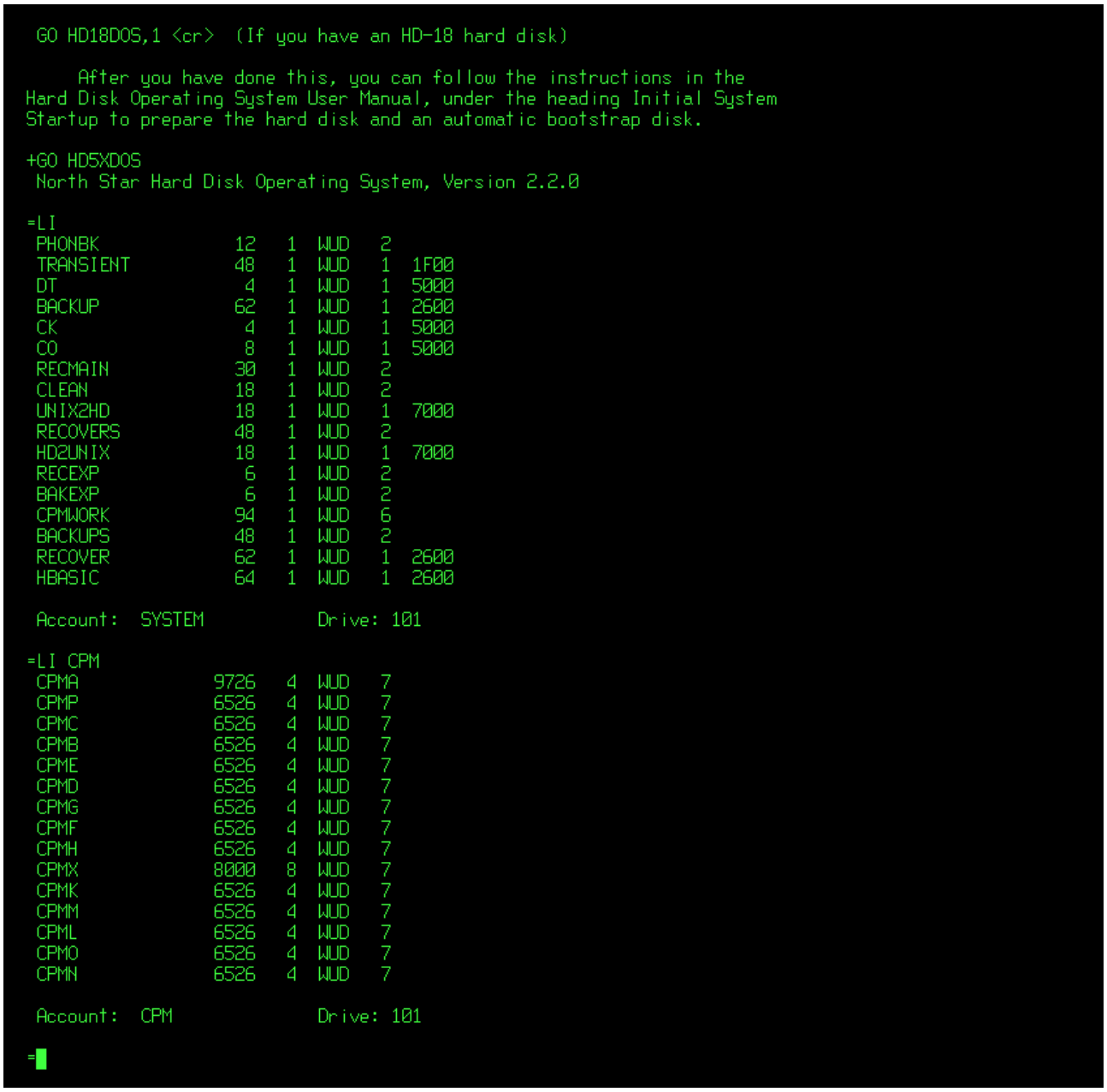

Fig 4. HDOS running in a high screen format.

1.8 NSE COMMAND-LINE START-UP OPTIONS

nse [-s] [-c config-file] [-m or -t]

-c <config-file>

Use an alternate config-file instead of the default '/home/username/nse/nse.conf' file. The alternate file should also be placed in the '/home/username/nse' directory.

-s

Use the single-density controller.

Note that the North Star single-density controller was not able to boot double-density disks and vice-versa. You must use the -s option if you are going to boot from a single-density disk-image.

-m

Specify the use of a display 96 characters wide by 36 lines.

-t

Specify the use of a display 80 characters wide by 25 lines.

1.9 NSE COMMAND-LINE STARTUP EXAMPLES:

nse -s -c nsdos.zzz

Start NSE using the single-density controller, booting from the single-density disk-image file which is specified in the '/home/username/nse/nsdos.zzz' configuration file.

nse

Start NSE using the double-density controller, booting from the disk-image specified in the default configuration file.

1.10 GETTING THE EMULATOR'S START-UP CONFIGURATION

1.10 THE WORK DIRECTORY

All of the emulators do their work in a directory which has the general look of

/home/username/emulator_name/

thus if user 'fred' is working with the 'nse' emulator, the nse work-directory is installed at

/home/fred/nse

In this work-directory will be found any logfiles, such as the **screenlog** which contains a record of all the output that was displayed by the screen during the emulator's activity. If any debug logging was required there will be a debug log called **xlog** written into the work-directory.

Also found in the work-directory are any configuration files which specify which floppy and hard disk images will be used while the emulator is working.

Subdirectories in the work directory are **disks**, **documentation**, and **info**.

The **disks** subdirectory is where floppy and hard disk images may be found. It is a good idea to put any other disk-images in there also. That is the first place that the emulator will usually look for disks.

The **documentation** subdirectory is where official North Star Computers documentation is found.

The **info** subdirectory is where other useful information may be placed.

1.11 CONFIGURATION FILES:

A user's default configuration file is found at

/home/username/work_directory/emulator_name.conf

thus user **fred** will find his *default* **nse** configuration file at

/home/fred/nse/nse.conf

Bear in mind, though, that any other configuration filename can be specified on the command-line by using the **-c** option, as in

/home/fred/.local/bin/nse **-c nse-nsdos.conf**

or even just simply

```
nse -c nse-nsdos.conf
```

if nse is located in one of your $PATH directories, and nse-nsdos.conf is located in the work-directory.

1.12 USER CONFIGURATION FILES: nse.conf

This is an actual configuration file.

```
###
### Configuration File for North Star Horizon Emulator (c) 200224
###
###             Avoid Editing This File Manually.
###
### Any Changes You Make Are Liable To Be Overwritten at Any Time.
###

fd1           /home/jvs/nse/disks/HDOS22BOOT.NSI
fd2           /home/jvs/nse/disks/HDOS22REC.NSI
fd3
fd4
hdd0          /home/jvs/nse/disks/hrz_cpm_30mb.nhd
hdd1
disk_dir      /home/jvs/nse/disks/
hd_delay      off
capslock      on
s2_in         /home/jvs/nse/serial2_in
s2_out        /home/jvs/nse/serial2_out
pl_in         /home/jvs/nse/parallel_in
pl_out        /home/jvs/nse/parallel_out
=======
log           /home/jvs/nse/xlog
screenlog     /home/jvs/nse/screenlog
debug_level   0000
break_addr    0000
break_on      off
trap_addr     FFFE
trap_on       off
```

Note that although there is provision for 4 floppies and two hard drives, this config file only specifies 2 floppies and one hard drive. The hard drive named can hold 14 quite large virtual disks out of the 16 CP/M drives possible, and so leaves just 2 drive letters for the floppies to use.

The **disk_dir** indicates where the last disk used was located, and where the emulator will look first for any other disks that are wanted.

The **hd_delay** is a way of slowing down the emulator's hard drive when changes are wanted to be made to the virtual CP/M drives as stored on the hard drive as HDOS files. Apart from that situation, the hd_delay can be left off as the normal case. See section XX for adjusting this.

Capslock is as described. Most people using NSDOS or CP/M will want the capslock on, but won't want it on for their host machine. See section XX to vary the condition.

All the items under the **=======** separator are normally used only during the development of the emulator itself and so will rarely be used (if ever) by most **nse** emulator users.

NOTE: While it is possible to edit the configuration file manually, your changes will be overwritten when any of the 'Options' or 'WorkShop' menu items are used.

2. Obtaining and Building 'NSE'

2.1 Linux Libraries required

Very few Linux libraries are required, apart from the standard packages installed on most Linux Distros.

The GUI Toolkit used is GTK+ Version 3, apart from a few deprecated functions from GTK+ Version 2.

This Toolkit can be installed using your Package Manager. If you're using one of the Debian derivatives such as Debian itself, Mint, or Ubuntu, this can be done by installing **libgtk-3-dev** and **libglib2.0-dev** using Synaptic or even just

sudo apt install libgtk-3-dev libglib2.0-dev

from the command line.

2.2 Get the source files

Download the NSE source code from https://itelsoft.com.au/code/nse_latest.tar.gz and move it to any convenient work directory. Untar and decompress the tarfile:

tar xvfz nse_latest.tar.gz

This will produce a subdirectory called nse. Move there.

cd nse

Compilation should be initiated with a simple **make** on the command-line**.**

If all goes well and the compile completes successfully, install the nse package with

make install

This will install the package in the **/home/*username*/nse** work directory. So user 'fred' will find a directory called **/home/fred/nse**.

A launcher icon will appear on the username's Desktop. Clicking on that should launch the emulator. It can be 'Drag n Dropped' to the Desktop Panel. Alternatively, NSE can be invoked from the command-line if the **nse** executable file is to be found somewhere within your $PATH list.

nse

2.3 What's in the /home/username/nse work-directory?

The /home/username/advantage directory has several important files:

nse.conf the default configuration file for NSE which holds most of your personal preferences which:

designates which CP/M disk image-files are mounted.
specifies what I/O files will be attached to the Horizon I/O ports.
preferred settings for capslock, hard-drive 'speed'.
preferred nse-development settings.

nse-nsdos.conf similar to above but loads NSDOS instead of CP/M

Avoid editing the nse.conf file manually. It gets updated automatically every time you make different choices on the Options and WorkShop menus, and will hold those settings indefinitely over more than one session.

pio_out destination of text from the parallel-out port: the 'LST:' device in CP/M

sio_out destination of text from the serial-out port: the 'PUN:' device in CP/M

2.4 Starting up NSE

Starting NSE can be done from the Desktop with one of the emulator icons or from the command-line. On start-up, the program will show the title (splash screen) and will then wait for user input. Usually, the user will then just hit the 'go' button because the installation process also provides the default configuration file, **nse.conf**, which will be found in the NSE top directory, **/home/username/nse**

nse.conf contains the default settings which are expected by the North Star Horizon computer:

A boot floppy *must* be in 'floppy drive' 1 at the minimum.

Several NSE settings are also stored in the nse.conf file. Such as Capslock ON/OFF, and whether the hard drives runs FAST or SLOW. The **nse.conf** file should not be edited manually. While that can actually be done, any changes you make may not be permanent.

If for some reason, the default configuration is not present in the top directory, then a new configuration file needs to be made. This is simply done by providing the user's settings with the 'Options Menu', and/or the WorkShop menu. See Section 3, page 18. Any time a setting is altered with these two menus, the new setting is saved automatically into the **/home/username/nse/nse.conf** file.

'Options Menu': Things to be changed by the everyday user.

'WorkShop' menu: Settings for use during NSE development. Most users won't need to bother with these.

2.5 Running North Star Horizon CP/M. The 'go button.

Now hit the 'go' button. The screen will clear, followed almost immediately by the CP/M Banner

```
64K CP/M vers 2.2 Horizon rev 1.2.0 HQ
Product of North Star Computers,Inc.
```

and then followed by the hard disk boot

```
Hard Disk Boot In-Process
To review connections, enter Semicolon (;) within a second or two
```

If no input (if you don't hit the ';' key in the next second or so), the hard-disk boot will display

```
No ; entered, so proceeding with prior connections
```

and it should then continue to the

```
A>
```

prompt when it will wait for a normal CP/M command-line as user input.

2.6. Pausing the Emulator. The 'pause' button.

In most cases you won't usually need to use the pause button unless things happen to move too fast for you, for instance to change floppies before the software moves on. Otherwise, using the Emulator is just like using a normal computer.

2.7 Rebooting/Resetting the Computer. The 'reset' button.

Just like the real thing, a reset will reboot the emulator from scratch. Use this button sparingly, your work may be lost.

2.8. Finishing Up. The 'exit' button.

Pack it up and put it away. The NSE program closes down and the emulator window is closed. Settings in nse.conf will remain for next session.

2.9 The ‘Status’ window

In between the two pairs of buttons, left and right, is a small window which displays short one-line messages. This is used to show information or warnings regarding the progress of the emulator. A short beep may be heard when some messages are shown. Examples:

North Star Horizon Emulator is RUNNING
Capslock is now ON
HARD DISK </home/fred/nse/disks/hrz_cpm_30mb.ndh> 29376 K
North Star Horizon Emulator RESET. Rebooted.
New Floppy “/home/fred/nse/disks/newflop.nsi” Created

3.0 The Options Menu

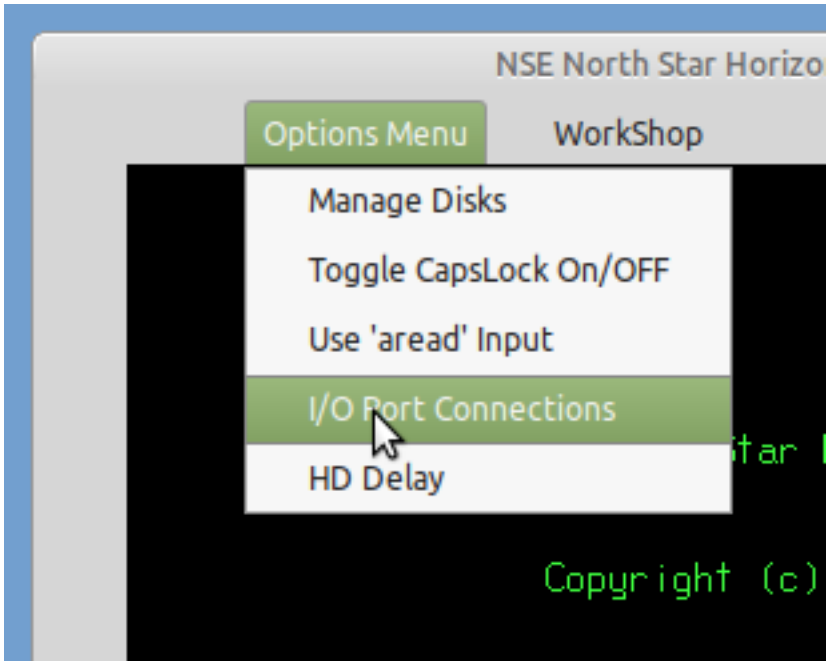

Fig. 5. The Options Menu

3.1 Disk Management.

The Disk Management menu item allows the user to ‘eject’ floppies and hard drives from the Emulator. The first displayed window shows what floppy disks happen to be ‘inserted’ in Floppy 1 or in Floppy 2. It also shows which hard drive was installed when the Emulator was booted.

Each disk-drive has two buttons: a ‘Change’ button which will install a different floppy-image or hard-disk image. And an ‘Eject’ button which removes any image-file which was previously installed.

If the ‘Change’ button is hit, a file-chooser dialog window opens and allows the user to browse through the whole file-system looking for a floppy-disk image to install. Once the file is selected, hit the ‘Select’ button to confirm your choice. The file-chooser window will close, the floppy-image is ‘inserted’ into the selected floppy-drive and is then ready for use.

The directory which the floppy-image came from will be used as the default disk directory in future disk-image searches. For this reason it is handy to store all your North Star Horizon floppy and hard-drive image-files in one or two directories.

A fourth option in the Disk Management window will enable the creation of a new floppy-disk image. That new floppy-image can then be ‘inserted’ into Floppy1 or into Floppy 2 using the ‘change’ option as above. It is recommended that floppy images have the file extension of ‘.nsi’ or ‘NSI’

If the new floppy-disk’s name is not recognised to be an absolute filename (Absolute filenames start with a ‘/’, as with a filename like /tmp/mynewfloppy.nsi) it will be recognised as relative filename, and created relative to the default disk directory. For instance if the default disk directory is /home/fred/nse/disks and the newly created floppy’s name is entered as mynewfloppy.nsi, the full absolute filename created would work out to be /home/fred/nse/disks/mynewfloppy.nsi

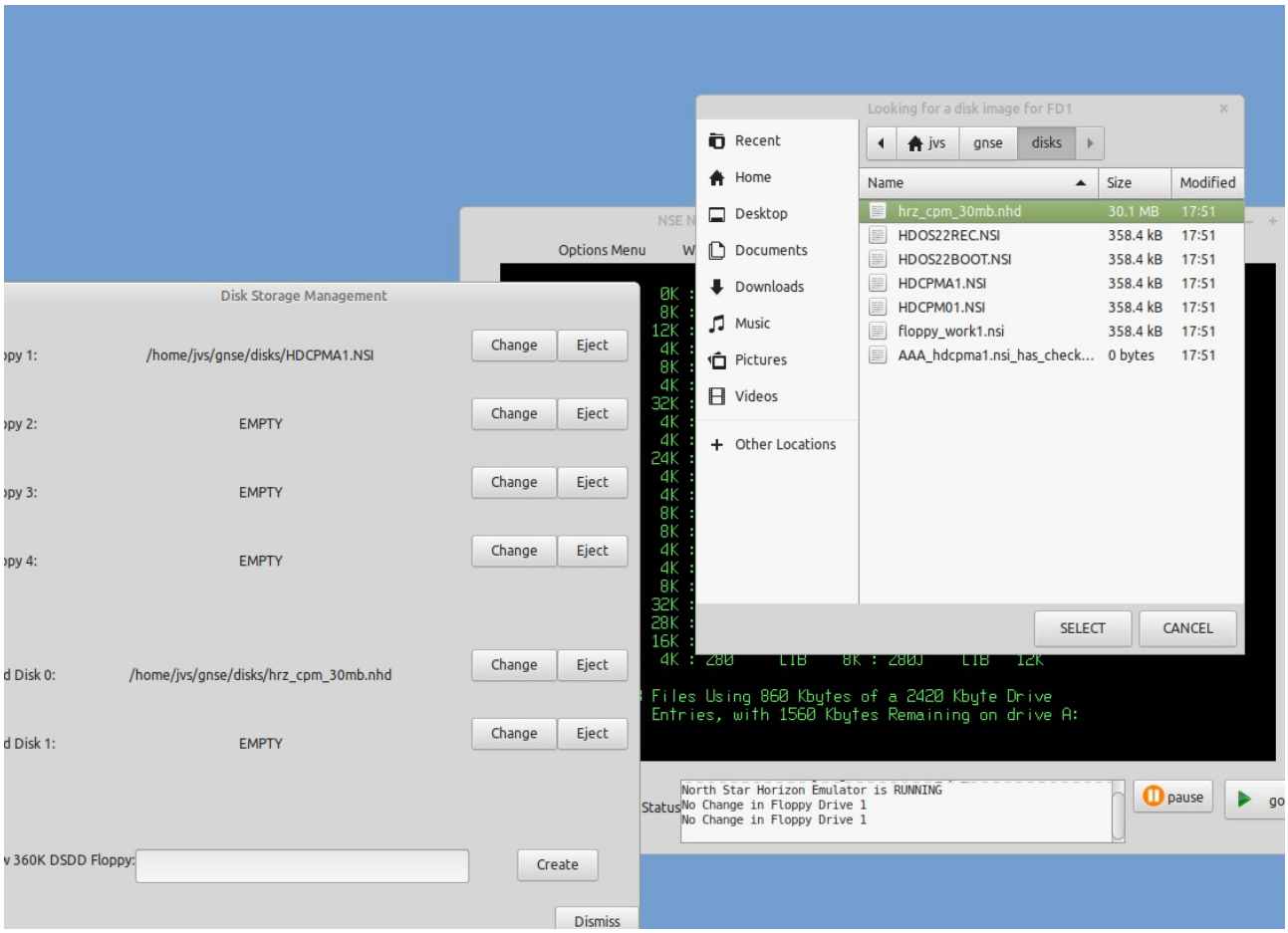

Fig. 6. Disk Selection Pop-Ups

3.2 Toggle Capslock ON and OFF

Many of the older Operating Systems will not recognise the use of lower-case characters. While one can use the actual Caps Lock Key to turn on the CapsLock, it would also turn on upper-case for the host Operating System as well. This can be a nuisance.

North Star DOS only understands uppercase commands, so it's necessary to toggle Capslock ON when using DOS. CP/M automatically converts command-line lowercase to uppercase anyway, so the Capslock setting can be set to personal preference.

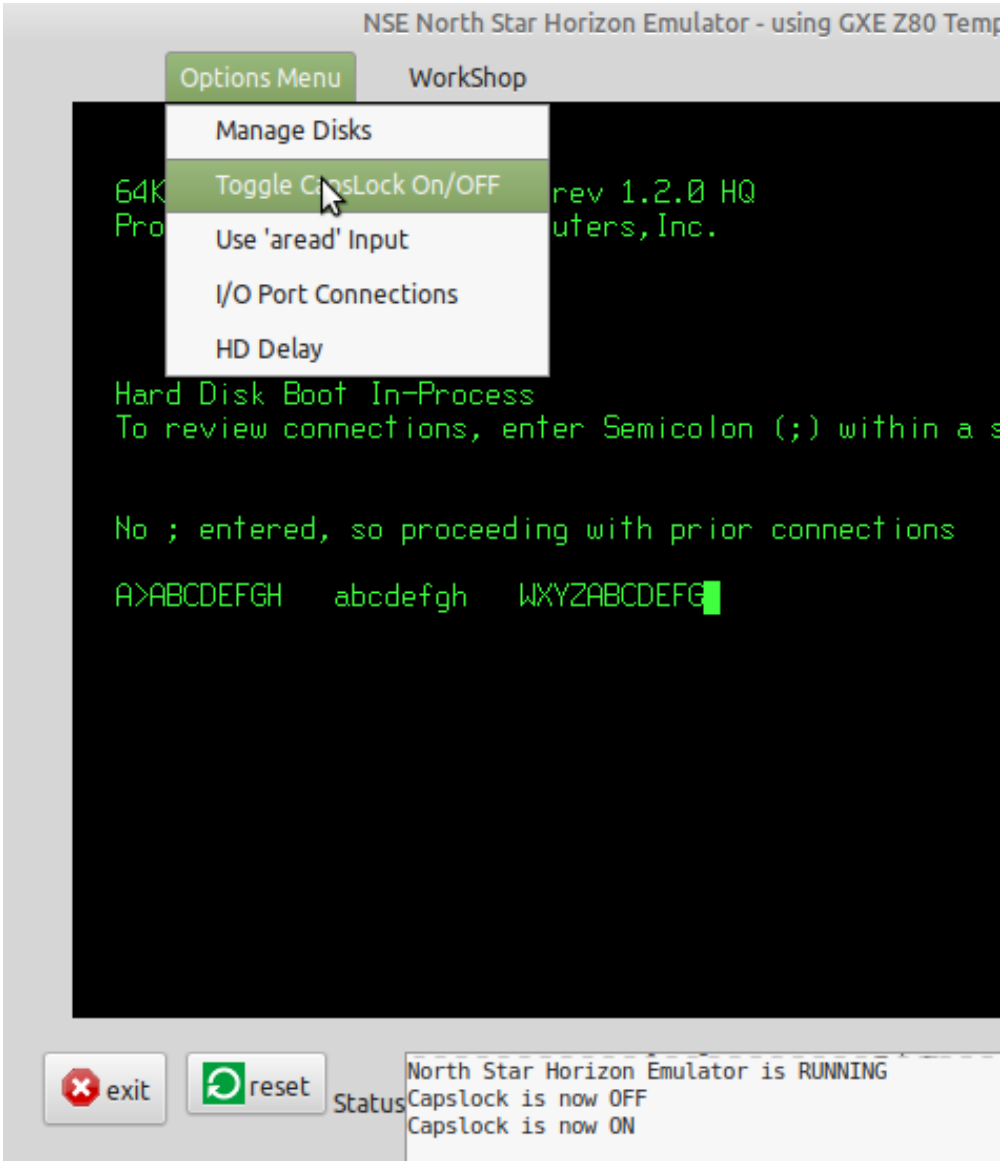

Fig 7. Capslock Toggled ON/OFF

In the screenshot above, Capslock starts out as being ON, showing the 'ABCDEFGH' in uppercase. Then Capslock is toggled to OFF, as shown in the Status Window, with the next set of characters being lowercase 'abcdefgh'. The Capslock is then toggled back ON, again showing in the Status Window, and the final 'WXYZABCDEFG' is again uppercase

3.3 Use ‘aread’ Input.

Read in an ASCII file from disk instead of having to type it all in manually. The ASCII file is read in line-by-line until it has all been entered. The keyboard then waits for user input, as it does normally

The Input File is selected with a file-chooser window. It is read in immediately after being selected.

Files read in with ‘aread’ will be processed in exactly the same way as they would if typed in at the keyboard. Excessively long lines will be rejected by the command-line processor of some operating systems, WordStar can ‘choke’ temporarily because it is unable to keep up with the faster input, but it usually recovers well.

3.3 Toggle HD Delay ON/OFF

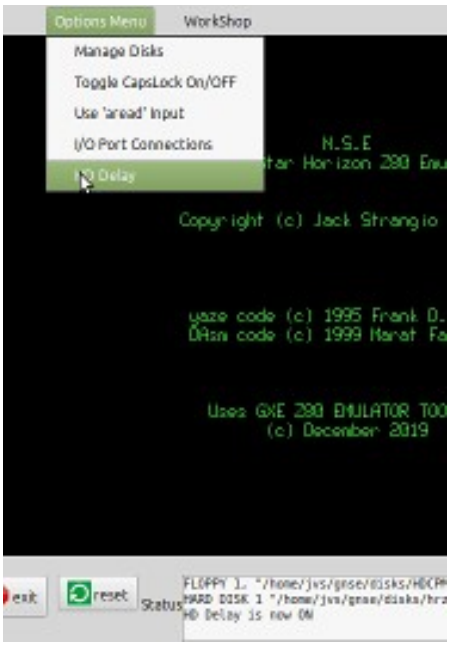

Fig 8. Status Window shows Hard Drive Delay Toggled ON (= SLOW).

NSE's 20 x speed emulation of the floppy disk drives and the hard disk drives is extremely fast! And the emulated 'correct speed' is still about 5 times faster than the REAL HD 'correct speed'. So the hard drive is deliberately slowed down even more so that the period allowed for entering a ';' to enable editing of the CP/M hard drive configuration is increased from about half a second to about 3 seconds. It is suggested that when you need to adjust the CP/M virtual-disk configuration, that the 'normal/slow' speed (HD Delay ON) is toggled on at the NSE start-banner screen, then the 'fast' speed (HD Delay OFF) is toggled back on, and left on, at the 'A> ' prompt. The default speed is 'Fast'.

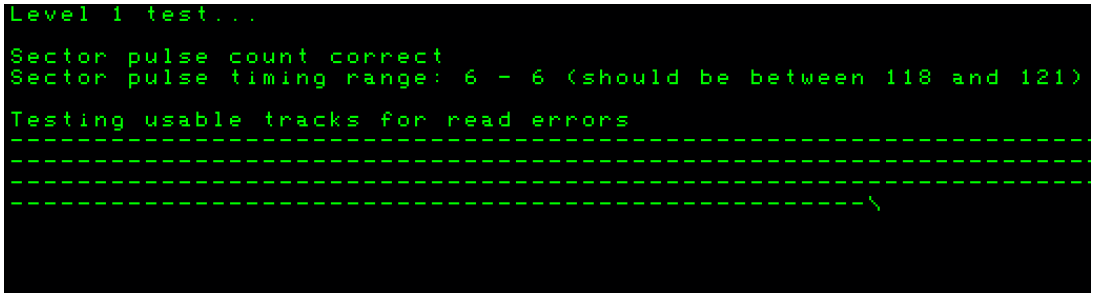

Fig 9. The hard drive (HD Delay is OFF) is 20 times faster than the 'correct speed'. (=FAST). The Level 1 Test on the HD Supplement Disk shows the pulse timing range is only 6 rather than 120.

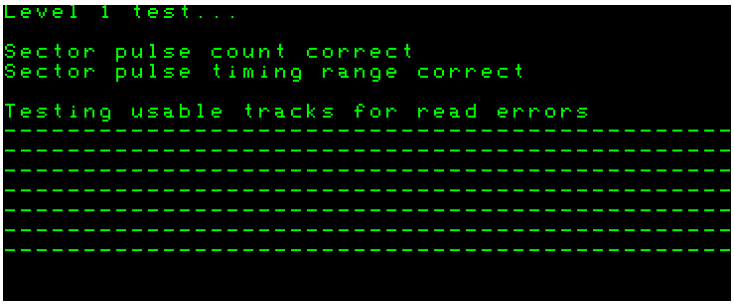

Fig 10. With HD Delay ON, (= SLOW) pulse timing range shows as "correct"!

It is very noticeable if the HD Delay is toggled SLOW/FAST while the HD Supplement Disk hard drive tests are in action.

3.4 Allocate I/O Port Files

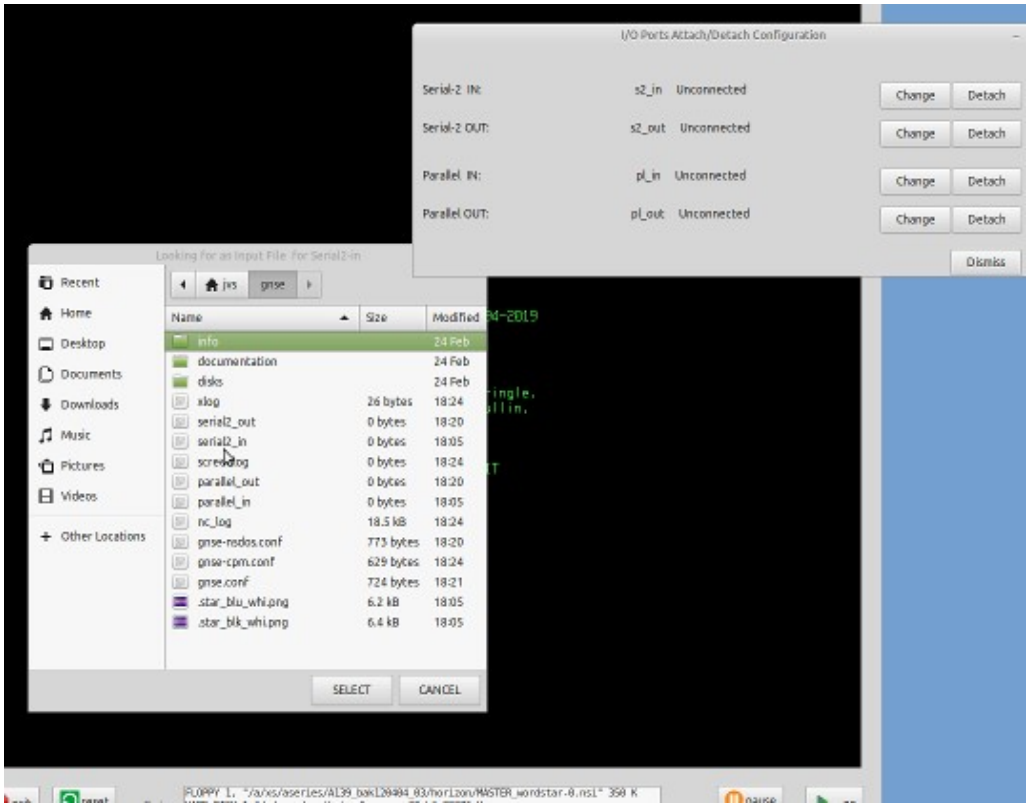

Fig 11. Allocating I/O Files to the Horizon I/O Ports

Attach or detach a unix file to or from a Horizon I/O Port. There is a parallel I/O port. And there is a second serial I/O port. In unix, everything is a file so one unix file or pipe is attached to the second serial-in port, and another to the second serial-out port.

Example: The ‘List’ device is allocated to the Horizon parallel port. Anything sent to the ‘List’ device will therefore show up as data in the file attached to the parallel output port.

3.5 TEXT COLOR OF THE EMULATOR OUTPUT

A selection of colors is available for the screen display. As this is pretty much a ‘set and forget forever’ option, it was decided against having a color-selection window as one of the ‘Options Menu’ items. To make a change it is simply a matter of selecting suitable values for the 24-bit RGB components, RED_LEVEL, BLUE_LEVEL, GREEN_LEVEL in the ‘gxe.h’ file and recompiling. Some examples -

Green on Black: (as default)

RED_LEVEL	0x3F
GREEN_LEVEL	0xFF
BLUE_LEVEL	0x3F

Amber on Black:

RED_LEVEL	0xFF
GREEN_LEVEL	0xBF
BLUE_LEVEL	0x3F

Yellow on Black:

RED_LEVEL	0xFF
GREEN_LEVEL	0xFF
BLUE_LEVEL	0x3F

White on Black:

RED_LEVEL	0xFF
GREEN_LEVEL	0xFF
BLUE_LEVEL	0xFF

4.0 NSE DEVELOPMENT ASSISTANCE

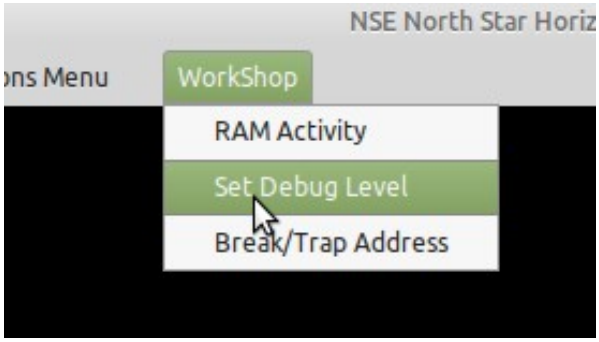

Fig 12. NSE Development menu: 'WorkShop'

4.1 Display RAM in the North Star Horizon virtual machine.

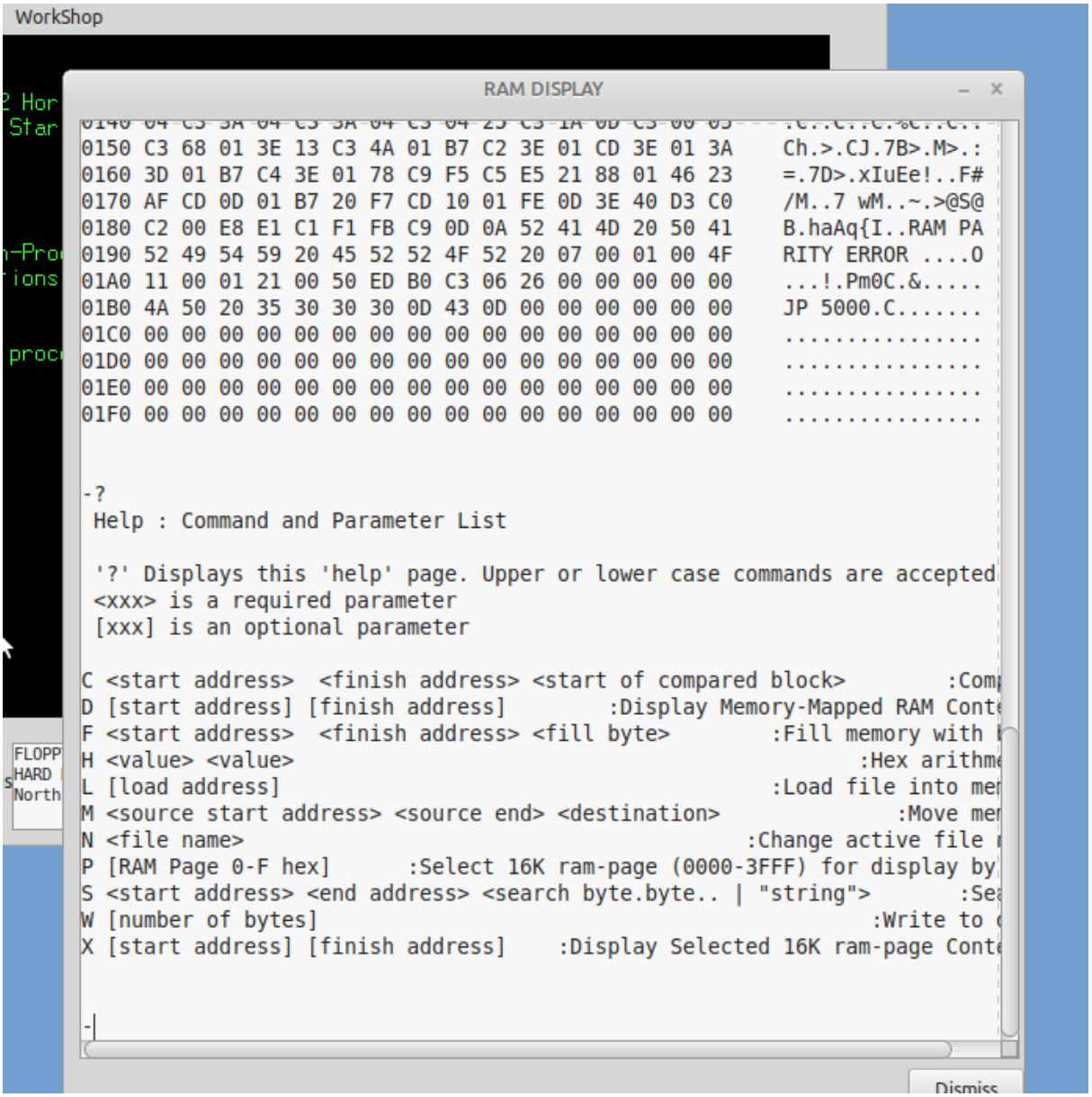

Fig 13. Display RAM Dialog

This subsystem has usage similar to CP/M 'DDT' or MSDOS 'DEBUG' Commands.

Upper or lower case commands are accepted

<xxx> is a required parameter, [xxx] is an optional parameter

compare
C <start address> <finish address> <start of compared block>
c 1a00 2000 2a00
Compare two equal-length blocks of memory. Only the bytes which are different will be displayed with location and values.

display
D [start address] [finish address]
d 0 12FF

Display the block of memory selected, showing bytes as hexadecimal and ASCII. If no start and end address specified, the command will continue for 100 H bytes from where it ended last.

examine/substitute
E <start address>
E 2CFF

Examine/change values at memory locations. The operation is stopped when no new value is entered, just a plain 'enter'.

fill
F <start address> <finish address> <fill byte>
f 1000 2000 55

Fill a block of memory with byte-value specified by <fill byte>.

hex
H <value> <value>
h 1267 abcd

Hex arithmetic results of the addition of two values and the subtraction of the second value from the first value.

load
L [load address]
l 2a00

Load the file (previously specified by the 'N' command) into memory. If a load-address is not specified the file will be loaded into location 0000 H.

move
M <source start address> <source end> <destination>
M 4d00 5000 6d00

Move the block of memory specified by the block's start and end into memory beginning at the destination address.

name
N <file name>
N xtest.bin.bas

Change active file-name which specifies which unix file will be used for 'load' and 'write' operations.

quit
Q
Quit from the RAM display subsystem back to the emulator's control console.

4.2 Setting the Debug Parameters for the ‘xlog’ Debugging File Output

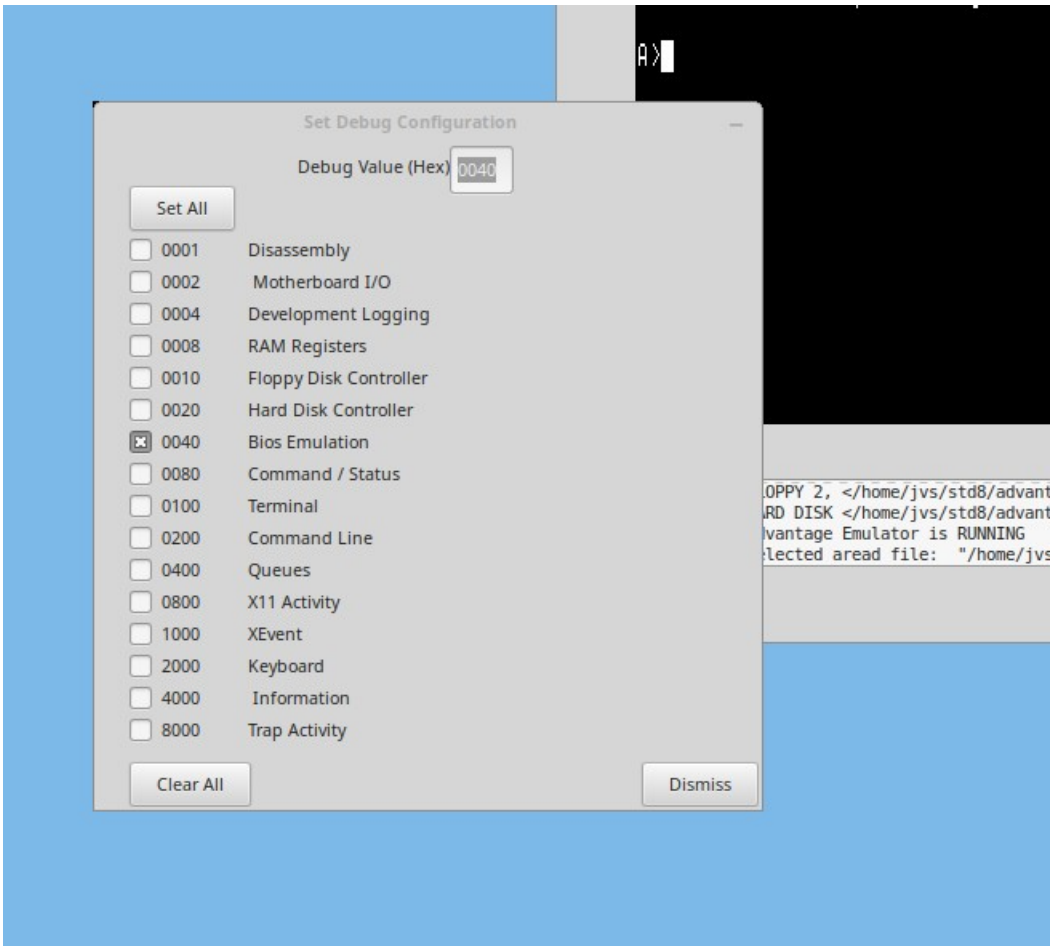

Fig 14. Setting Debug Logging Parameters

Depending on which items are selected for debug logging, *a lot* of logging output can be produced. Take care that your filesystem does not get over-filled.

4.3 Setting Execution Breakpoint Address, and Trap Address

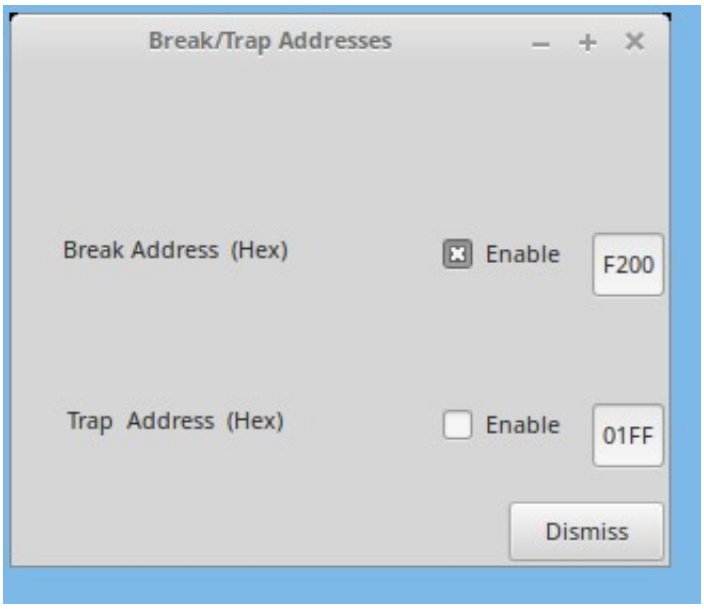

Fig 15. Enabling and Setting Break And Trap Addresses

break

Set a breakpoint address to stop the emulator at a pre-specified address. This is equivalent to the “PAUSE” button, but it occurs at a desired execution address. The contents of the RAM can then be examined by using the Ram Display functions. Hitting the ‘go’ button will resume execution from that breakpoint address and it will continue until that breakpoint address is again reached, unless the breakpoint is disabled while execution is stopped.

trap

Set a trap address to stop the emulator, perform a user-specified unix operation, return to the emulator and continue.

A dummy ‘trap’ function is included in the emulator source (trap.c) which merely prints the trap address and the register values. The trap function could be used to access parts of the host unix system or perform any other required operation.

Both the ‘break’ and ‘trap’ functions are enabled and disabled by the Check Buttons associated.

4.4 Booting with Either the Single-Density or Double-Density Floppy-Disk Controller

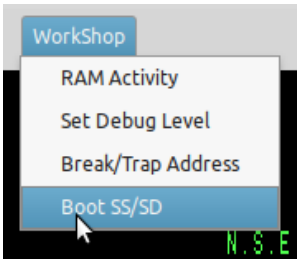

Fig 16. Changing from the Single-Density Floppy Controller to the Double-Density Floppy Controller

As with the actual Horizon, you can boot up with a single-density floppy or a double-density floppy. Also in keeping with the real Horizon, you would be required to to change the Floppy Disk Controller itself. The way to do that in NSE is to use the menu item in the WorkShop menu to set to boot the floppy-type that you want.

In effect, you are toggling whether you will be using the original Single-Sided/Single-Density or the later Double-Density controller. Note that the Horizon was unable to boot a single-density disk using the double-density controller, and was unable to boot or read a double-density disk using the single-density controller.

While NSE will allow you boot using boot-disks of both densities, you will have to either hit the 'RESET' button after toggling the two controllers, or even switch off NSE and restart it. The sequence is

Insert a boot disk with a different density from previously.
Toggle the WorkShop->Boot SS/SD menu item to match the floppy's density.
Hit the 'RESET' button OR exit NSE then restart it. ('RESET' is simpler.)

4.5 Log the debug information to Unix Disk File.

Automatically sends debugging/information output to the ‘xlog’ unix file.Take care, because the quantity of information sent to the log file can reach the maximum size (2 Gig in 32-bit systems, whole disk or whole filesystem in 64-bit systems) within a fairly short time.

Unless you're doing development on the North Star Horizon Emulator itself, it probably will not be useful to use any debug logging at all.

4.6 Log the Screen Output to Unix Disk File.

Automatically sends all ASCII screen text output to the ‘screenlog’ unix file. This can be handy to refer to if text output scrolls off the top of the screen before you can read it.

5. HELPER PROGRAMS

5.1 North Star Tools

These are placed in the user's personal **bin** directory, /home/username/.local/bin, the current default location of a user's executables, and can be called directly from the command-line.

5.2 **mkhd** (make hard-disk-image file)

mkhd is used to produce NSE hard-disk image files. The smallest of the images of the North Star 'standard' hard-disk types (as included in the HD5XTEST program) is 5 megabytes, the largest is 30 megabytes.

A typical example session with mkhd is shown (user input in **bold**):

```
centrepoint [jvs] /home/jvs/nse/disks > mkhd


                        === mkhd ===
                        Version  3.2

Prepares a "Standard" 5-inch Hard-Disk Imagefile for use with North Star
Horizon Emulator (nse) and Advantage Emulator (ade) running HDOS.

Disk-image sizes available range from 5 MB to 30MB.

                                 Usable    Usable  Shipping  Total   Total
No.   Type   Rev. Cylinders Heads Sectors  Capacity Cylinder Sectors Capacity

 1    SG5A   1.0    153      4     9792     4.90 M    153      9792   4.90 M
 2    TN5A   2.0    153      4     9792     4.90 M    153      9792   4.90 M
 3    MS5B   2.0    306      2     9792     4.90 M    336     10752   5.38 M
 4    RD5B   2.0    306      2     9792     4.90 M    319     10208   5.10 M
 5    SG5B   2.0    306      2     9792     4.90 M    306      9792   4.90 M
 6    TN5B   2.0    306      2     9792     4.90 M    306      9792   4.90 M
 7   CM10E   2.0    612      2    19584     9.79 M    650     20800  10.40 M
 8   MS10E   2.0    612      2    19584     9.79 M    656     20992  10.50 M
 9   CM15C   2.0    306      6    29376    14.69 M    306     29376  14.69 M
10   SG15C   2.0    306      6    29376    14.69 M    306     29376  14.69 M
11   RD15C   2.0    306      6    29376    14.69 M    319     30624  15.31 M
12   TN15C   2.0    306      6    29376    14.69 M    306     29376  14.69 M
13   MS15D   2.0    480      4    30720    15.36 M    522     33408  16.70 M
14   MS15E   2.0    459      4    29376    14.69 M    522     33408  16.70 M
15   CM20E   2.0    612      4    39168    19.58 M    650     41600  20.80 M
16   MS20E   2.0    612      4    39168    19.58 M    656     41984  20.99 M
17   RD20E   2.0    612      4    39168    19.58 M    639     40896  20.45 M
18   MS30D   2.0    459      8    58752    29.38 M    522     66816  33.41 M
19   CM30E   2.0    612      6    58752    29.38 M    650     62400  31.20 M
20   MS30E   2.0    612      6    58752    29.38 M    656     62976  31.49 M
21   RD30E   2.0    612      6    58752    29.38 M    639     61344  30.67 M


   Select ( '0' to exit) : 1


     Type: SG5A disk:   4.90 M usable capacity.   ---- Is that correct? y

    creating disk-image type SG5A,  4.90 M.

    Enter file name for this disk: /tmp/horizon5mb

Disk ImageFile:  /tmp/horizon5mb    requested.


Disk ImageFile: '/tmp/horizon5mb'   created OK.

Creating SYSTEM account. Do you want to include the TRANSIENT file? (Y/n) y

 TRANSIENT for the Advantage, or the Horizon?            (a/H) h
  Horizon TRANSIENT installed.

Done.
```

I suggest the use of the .NHD extension for these North Star Hard-Disk Image files. This extension,

like most, is probably already in use elsewhere but is unlikely to be confused with our usage.

5.3 __nshdls__ (nshd list directory)

nshdls   <North Star hard-disk-image>

nshdls disks/SG5A-1.NHD

nshdls displays lots of information from the file-entries in the North Star Hard Disk index. Because the position of the file-entries within the directory depends on a hash function, the directories always seem jumbled – this was a trade-off way back in the days of slow hard-disks.

There are two options to **nshdls**:

-d   Show deleted files as well as existing files
-l   (long) Show more details about the directory and hard drive

```
centrepoint [jvs] /home/jvs/wrk/nse > nshdls -d disks/cm10ex.nhd


FileName  Size:Sectors---Blocks     Account    Filetype       Description


BUSINESS          0       0 blk     - - -         0     ACCOUNT (02) "BUSINESS"
CPMA           2031    4062 blk     SYSTEM        7     CP/M Virtual Disk File
TRANSIENT         0       0 blk     SYSTEM        1     NS **  DELETED  **
CPMB           2031    4062 blk     SYSTEM        7     CP/M Virtual Disk File
CPMC           2031    4062 blk     SYSTEM        7     CP/M Virtual Disk File
CPMD           2031    4062 blk     SYSTEM        7     CP/M Virtual Disk File
CPME           2031    4062 blk     SYSTEM        7     CP/M Virtual Disk File
CPMF           2031    4062 blk     SYSTEM        7     CP/M Virtual Disk File
CPMG           2031    4062 blk     SYSTEM        7     CP/M Virtual Disk File
CPMH           2031    4062 blk     SYSTEM        7     CP/M Virtual Disk File
SYSTEM            0       0 blk     - - -         0     ACCOUNT (01) "SYSTEM"
CPMWORK          47      94 blk     SYSTEM        6     CP/M Workfile
LEDGER          250     500 blk     BUSINESS      3     HBASIC data file
```

5.4 __nshdbm__ (nshd bitmap)

nshdbm <North Star hard-disk-image>

nshdbm disks/SG5A-1.NHD

nshdbm gives a slightly more compact set of information than the long North Star Hard Directory Listing and also shows a graphical representation of the usage of the hard disk's file-space.

```
 /tmp/nse_120805/disks > nshdbm SG5A-1.NHD


          Bitmap and Directory for North Star Hard Disk Image "SG5A-1.NHD"  (4.896 Megabytes)

Total DIBs = 612 (maximum usable = 602, as also uses 1 DIB for System Track, 8 DIBs for File-Directory, 1 DIB for Test Track)
                            [ 1 DIB is 16 sectors unless a power-of-2 factor is applied ]


Dir    Dir             File            FileSize    Bytes  dib/dib    ndib0    ndib0     File  Type       File Type
Offset Type  Accnt  --  Name  -- Attr Sectors Blocks inLast Num/Sects Address Img_Offset  Type  Data       Description
___________________________________________________________________________________________________________________________
011A00  1    0001  CPMA          80    1562  3124    200    6  64     0020   00040000     7   0001.00  CP/M Virtual Disk File
012000  1    0001  TRANSIENT     80      24    48    200    4  16     0003   00006000     1   1F00.00  Executable, TypeData=GO
012A00  1    0001  CPMB          80    1562  3124    200    6  64     0084   00108000     7   0001.00  CP/M Virtual Disk File
013400  1    0001  DT            80       2     4    200    4  16     0005   0000A000     1   5000.01  Executable, TypeData=GO
014400  1    0001  BACKUP        80      31    62    200    4  16     0006   0000C000     1   2600.00  Executable, TypeData=GO
016E00  1    0001  CLEAN         80       9    18    200    4  16     0013   00026000     2   0010.00  HBASIC program
017000  1    0001  CK            80       2     4    200    4  16     0010   00020000     1   5000.01  Executable, TypeData=GO
017020  1    0001  CO            80       4     8    200    4  16     0011   00022000     1   5000.01  Executable, TypeData=GO
017400  1    0001  RECMAIN       80      15    30    200    4  16     0012   00024000     2   001D.00  HBASIC program
01A600  1    0001  RECOVERS      80      24    48    200    4  16     0014   00028000     2   002E.00  HBASIC program
01C400  1    0001  RECEXP        80       3     6    200    4  16     0016   0002C000     2   0003.00  HBASIC program
01CC00  1    0001  BAKEXP        80       3     6    200    4  16     0017   0002E000     2   0005.00  HBASIC program
01DE00  1    0000  SYSTEM        01       0     0    000    0   1     0000   00000000     0   0000.00  ACCOUNT directory entry
01DE20  1    0001  CPMWORK       80      47    94    200    4  16     0001   00002000     6   0000.00  CP/M Workfile
01E400  1    0001  RECOVER       80      31    62    200    4  16     001A   00034000     1   2600.00  Executable, TypeData=GO
01EC00  1    0001  HBASIC        80      32    64    200    4  16     001C   00038000     1   2600.00  Executable, TypeData=GO
01F200  1    0001  BACKUPS       80      24    48    200    4  16     0018   00030000     2   0030.00  HBASIC program


128 DIBs per line. 'S' = System, 'D' = Directory, 'X' = TestTrack, '.' = Unallocated, 'o' = Allocated, 'U' = Multiple Allocation

SoooooooDDDDDDDDoooooooooooooooooooooooooooooooooooooooooooooooooooooooooooooooooooooooooooooooooooooooooooooooooooooooooooooooo
oooooooooooooooooooooooooooooooooooooooooooooooooooooooooooooooooooooooooooooooooooooooooooooooooooooooo........................
................................................................................................................................
................................................................................................................................
...................................................................................................X
```

```
nullius [jvs] /tmp/nse/disks >
```

5.5 nshdcp (nshd copy file to unix)

nshdcp <North Star hard-disk-image> <Filename>

nshdcp SG5A-1.NHD HBASIC

nshdcp extracts a North Star HDOS file from the North Star Hard Disk image. The filename to be extracted is case-sensitive, although the huge majority of HDOS filenames are upper-case only.

Note that any CP/M files are contained within CP/M virtual disks which are large HDOS files. **nshdcp** will only extract the virtual disk file itself, rather than any indivual CP/M file contained within the virtual-disk file.

5.6 unskew-hd-image

unskew-hdimage <North Star Hard Disk Image> <unskewed image file>
OR
unskew-hd-image <unskewed image file> <North Star Hard Disk Image>

unskew-hd-image SG5A-1.NHD image-plain-a

unskew-hd-image can be dangerous to your hard-disk image-files. **Be careful!** It will be used mainly if you are trying to resurrect portions of files which have been lost by removing the interleaving of the sectors and giving a flat file with everything in correct order.

5.7 nsfilecalc (calculate filesizes in terms of NSDOS 256-byte 'blocks')

ns*filecalc*

```
nullius [jvs] /tmp/nse/disks > nsfilecalc

            North Star DOS/HDOS File-Size Calculator
                  copyright 2012 Jack Strangio

A North Star Floppy Disk file is restricted to a maximum length of 66
tracks on a DQ disk, or 660 sectors, 1320 blocks, 330 kilobytes.

A North Star Hard-Disk file is made from 'hunks' containing multiple
sectors.  These 'hunks' were originally so-named by North Star, but later
this name was changed to 'DIBs'.

Each DIB ('Data Incremental Block', similar to 'clusters', 'extents', etc.
in other operating systems) contains a multiple of 16 sectors.  There can be
a maximum of 128 DIBs per file.

Since this could really restrict the maximum size of a file, a power-of-2
factor can be applied to 16 giving 16, 32, 64, 128, or even up to 256 sectors
per DIB.  Consequently, it becomes possible to produce a file which can go up
to the maximum allowable file-size on a hard-disk: 65,535 blocks, 32,768
sectors or 16.384 megabytes.

Each file contains its own internal DIB-directory, which takes up the first
sector of the file itself.  Keep this 'loss' of the first file sector in
mind when creating your files on the hard-disk.  The Hard-Disk Directory (or
Index) merely tells HDOS where the file's first sector with its
DIB-directory is located upon the hard-drive.


 Bytes                                (1)
 North Star Blocks (256-byte)         (2)
 Hard-Disk Sectors (512-byte)         (3)
 North Star DIBs ('clusters','extents')  (4)
 Kilobytes (1024 bytes)               (5)
 Megabytes (1000x1024 bytes)          (6)


         Select Units: ('0' to quit)  6
                 Enter Value wanted : 3
```

```
File is:  3072000 bytes, 12000 blocks, 6000 sectors, 94 DIBs, allocation factor = 4, 3000.0 KB

    HDOS Command Line:     CR FILENAME[[,ACCOUNT],DISK_UNIT]  12000  4


****    That size of file has unused sectors in the last DIB.    ****

      If all sectors of the last DIB were to be included, the file's size would then become:

          3079680 bytes, 12030 blocks, 6015 sectors, 94 DIBs, allocation factor = 4, 3007.5 KB

    HDOS Command Line:     CR FILENAME[[,ACCOUNT],DISK_UNIT]  12030  4

nullius [jvs] /tmp/nse/disks >
```

nsfilecalc will notify you whether the file-size you have requested will not completely fill a DIB. For instance, if the disk space is being allocated in 64 block chunks, a file that's 65 blocks long will take up 128 blocks. So if you're making a CP/M virtual disk, it costs you no more to make your 'disk' have 128 blocks in size than a 'disk with only 65 blocks of disk space.

Therefore, if there is unused space left in the allocated disk area you may, if you want, increase the size requested up to the end of the last DIB. Hence the recommendation in the printout above of making a 12000-block disk into a 12030-block disk.

5.8 nsfd2u (copy NSDOS file from floppy-disk to unix)

nsfd2u <NSDOS disk-image>

nsfd2u D04B01.NSI

nsfd2u reads the files off a double-density North Star DOS disk image file and creates copies of those files in the unix file space.

The unix filenames will have the format of <Name of File>_<FileType>[_Go-Address]. The Go-address will only be used with a file of Type 1 (executable).

example 1.
The M5700 executable file is Type 1 and has a Go-Address of 5700 H; this has a unix file name of M5700_1_5700

example 2.
The BASIC program called OTHELLO is Type 2 (BASIC Program) and not being a executable Type 1 will have no Go-Address; this has a unix file name of OTHELLO_2

5.9 u2nsfd (copy file from unix to NSDOS floppy-disk)

u2nsfd <unix file> <NSDOS disk-image>

u2nsfd M5700_1_5700 MYDOSDISK.NSI

u2nsfd will copy a file from the unix file space onto a double-density North Star DOS disk image file.

If the above filename format (as in nsfd2u) is used for the North Star DOS filename in the unix file space, then the file will be added to the NSDOS disk directory complete with Type attributes and Go-Address if applicable. If the NSDOS directory already has a file of the same name, the new file will replace the earlier file.

If the above filename format is not used, the file-type defaults to Type 0 (undefined). This can then be altered using the TY command in NSDOS:

TY <filename> <File-Type> [Go-Address]

5.10 compact

compact <NSDOS disk-image>

compact MYDOSDISK.NSI

compact will 'compact' a North Star DOS disk image file. It will act similar to a defragmenting of the disk-image file by moving all files towards the beginning of the disk, eliminating any unused space between the files where previously deleted files once were. [Same as running the compact program in the emulator]

5.11 nsfdls (NS floppy-disk list directory)

nsfdls <*NSDOS disk-image*>

nsfdls MYDOSDISK.NSI

nsfdls lists the directory of the floppy-disk image file in the same format as the LI in NSDOS. [Same as running the LI program in the emulator]

5.12 mkfs.ns

mkfs.ns [-s] <disk-image filename>

mkfs.ns -s MYSSSDDISK.NSI

mkfs.ns creates an empty North Star DOS formatted floppy-disk image. It can produce either single-sided, single-density disk-images (88K) or double-sided, double-density disk-images (350K). The default size is 350K, if you use the '-s' option an 88K disk-image is produced. The first 8 characters of the filename are used as the disk-label. [Same as creating a new floppy in the Disk Manager screen]

5.13 jdz80 (Z80 disassembler)

jdz80 is a slightly improved version of Marat Fayzullin's 1999 DAsm, in which relative jump destination addresses are calculated and displayed rather than just displaying the relative jump offsets.

5.14 OTHER TOOLS

5.14 cpmtools

Life is simpler with cpmtools-2.7 (or later) which can be obtained from most linux repositories. This set of utilities can be used to copy files directly between North Star CP/M disk-images and the unix/linux file space. It will be necessary to add the following disk definitions to the cpmtools config-file **diskdefs** which is usually at /etc/cpmtools/diskdefs.

```
diskdef nsfd
  seclen 512
  tracks 70
  sectrk 10
  blocksize 2048
  maxdir 64
  skew 5
  boottrk 2
```

```
  os 2.2
end

diskdef nshd4
  seclen 512
  tracks 512
  sectrk 16
  blocksize 4096
  maxdir 256
  skew 0
  boottrk 0
  os 2.2
end
```

The added disk-definitions will enable cpmtools to understand the North Star CP/M disk formats, both the floppy-disk images and the larger CP/M Virtual Disk Images on the hard disk. (Note that you will need to copy the hard-disk CP/M Virtual Disk image-file off from the hard disk image-file by using the **nshdcp** program before you can start to use the cpmtools with it.)

The utilities in cpmtools include:

cpmls list files in the North Star CP/M disk-image
cpmcp copy files to and from the North Star CP/M disk-image
cpmrm delete files from the North Star CP/M disk image
mkfs.cpm prepare stub disk for CP/M. In my experience, this does not work properly.
Instead, use mkfs.ns to produce an NSDOS disk then FORMAT it for CP/M.

5.15 screenlog

screenlog is not a tool as such but a record of NSE's screen output.

5.16 xlog

xlog is not a tool but is a record of all debugging information. Can make very large log files.

6.1 OTHER FILES REQUIRED

Various floppy-disk image files:

These are available from various sources. Most of them have a .nsi extension.

6.2 COMPILING LIBRARIES REQUIRED

The linux libraries required are GTK+ version 3

6.3 VARIOUS USEFUL MANUALS

Most of the manuals are available from http://www.hartetechnologies.com/manuals/Northstar/ or from https://itelsoft.com.au.

Probably the most useful are:

North Star DOS Rev 5
North Star BASIC Version 6
North Star Horizon Emulator (NSE) User Guide (this manual)
North Star Hard Disk Operating System Manual
North Star CPM 2.2 Manual
North Star CPM 2.2 Preface to the Addendum
North Star CPM 2.2 Addendum

These are all included in the 'documentation' directory

6.4 BUGS

I feel I have got many bugs out, which makes NSE very usable. But there are still a few to go, apart from the things that could be done to make NSE not quite so rough-edged. It certainly is not yet anywhere near as elegant as I would like, and the fault-lines between the several programs that NSE is based upon are still very visible. Please inform me of any bugs that you discover. Email me at: jackstrangio@yahoo.com

6.5 TODOs

More realistic emulation of Parallel and Serial I/O, particularly the in-ports. There is a list in the TODO file.

6.6 AUTHOR and SUPPORT

Jack Strangio <jackstrangio@yahoo.com>

Website: `https://itelsoft.com.au`

APPENDIX A.

HOW TO ENABLE NSDOS AND CP/M TO USE THE SECOND HARD DRIVE ON HD5X CONTROLLER

NOTE: The two North Star boot disks supplied with NSE have been adjusted so that the second hard drive is already configured in. This **may not** be the case with other boot disks.

1. NSDOS

I discovered that the 'NSDOS for HARD-DISK version 2.20' master disk (archive disk : D04B01.NSI) is configured by default to use only the first hard-disk. Then by logging the path of the flow in the hard-drive initialisation code I saw that an incorrect port-number for the second hard-drive was being used. (0xFF in place of 0x70).

I found that to use two hard-drive units, we need to enable use of second hard-drive by replacing an 0xFF byte at 0504H in memory by 0x70 (base port for second hard-drive in controller). If we load HD5XDOS into memory at 5000H then the relevant byte to alter is at 5404H.

Bytes 0503H & 0504H then become 60H & 70H. We also need to 'restore' the second hard-drive using the TOTREC software.

I did intend to remove that set of SYSTEM-account software on the second hard-drive, but it wasn't worth the effort. (Disk space is cheap). And besides, the **mkhd** program can install the SYSTEM account and the TRANSIENT program for you.

Sample session:

```
 North Star Hard Disk Operating System, Version 2.2.0

=ML
 TRANSIENT          48   1  WUD   1  1F00
 DT                  4   1  WUD   1  5000
 BACKUP             62   1  WUD   1  2600
 CK                  4   1  WUD   1  5000
 CO                  8   1  WUD   1  5000
 RECMAIN            30   1  WUD   2
 CLEAN              18   1  WUD   2
 <RECOVER.LIST>     56   1  WUD   3
 RECOVERS           48   1  WUD   2
 RECEXP              6   1  WUD   2
 BAKEXP              6   1  WUD   2
 CPMWORK            94   1  WUD   6
 BACKUPS            48   1  WUD   2
 RECOVER            62   1  WUD   1  2600
 HBASIC             64   1  WUD   1  2600

 Account:  SYSTEM           Drive: 101

=LI HDUNIT2,102

 Type: 125   Drive: 102   Sector:     2   Hard Disk Drive Not Found

=DH 0500-050F
 0500  C3 24 05 60  FF 00 00 00   28 00 03 00  01 00 08 07

=LF HD5XDOS,1 5000

=DH 5400-540F
 5400  C3 24 05 60  FF 00 00 00   28 00 03 00  01 00 08 00

=DS 5404
 5404 FF= 70

=SF HD5XDOS,1 5000

=
```

(Reboot here)

```
 North Star Hard Disk Operating System, Version 2.2.0


=LI HDUNIT2,102
```

```
 CPM-A:           4096   4  WUD   7
 CPM-B:           4096   4  WUD   7

 Account:  HDUNIT2          Drive: 102

=ML
 TRANSIENT          48   1  WUD   1  1F00
 DT                  4   1  WUD   1  5000
 BACKUP             62   1  WUD   1  2600
 CK                  4   1  WUD   1  5000
 CO                  8   1  WUD   1  5000
 RECMAIN            30   1  WUD   2
 CLEAN              18   1  WUD   2
 <RECOVER.LIST>     56   1  WUD   3
 RECOVERS           48   1  WUD   2
 RECEXP              6   1  WUD   2
 BAKEXP              6   1  WUD   2
 CPMWORK            94   1  WUD   6
 BACKUPS            48   1  WUD   2
 RECOVER            62   1  WUD   1  2600
 HBASIC             64   1  WUD   1  2600

 Account:  SYSTEM           Drive: 101



 CPM-A:           4096   4  WUD   7
 CPM-B:           4096   4  WUD   7

 Account:  HDUNIT2          Drive: 102

=DH 0500-050F
 0500  C3 24 05 60  70 00 00 00   28 00 03 00  01 00 08 07

=
```

2: CP/M

In a similar manner to HDOS, the CP/M floppy disk master is also configured to use just the first hard-drive.

In this case, we don't have the easy method of doing the required changes within the emulator itself. You will need to find a hex editor, such as my own 'uddt' or similar, so that the bytes within the floppy disk-image can be altered.

In the CP/M disk (archive disk: N2212_64.NSI), there will be 60H, FFH bytes at positions 1AC03H and also at 20F08H from start of floppy-disk image.

Change the bytes at 1AC04 and at 20F09 from FFH to 70H. Save the new values to disk-image.

APPENDIX B.

REPLACE THE COMMAND-LINE UNDERLINE IN HDOS 2.2.0 WITH A BACKSPACE

In the days of the Teletype, we made do with a back-arrow or underline instead of the destructive backspace which we are more comfortable with nowadays. It's a bit of a shock to the system when we have to go back to the 'bad old days' of the command-line underline.

METHOD ONE

Change the define in the nse.h file so that the variable WANT_DESTRUCTIVE_BACKSPACE is set to TRUE. This is the default for NSE.

METHOD TWO

This patch will change both the backspace and the underline to the destructive backspace, if you want to fix just the underline then only adjust the byte at XX1B H. If you only want the backspace to be fixed then just change the byte at XX1F H, as shown below.

Using a hex editor, load the D04B01.NSI floppy-disk file.

Change the two bytes at 3E1B H and at 3E1F H to point to the Control-H code at 3E5F H by changing the value of the byte at 3E1B H to 43 H and the value of the byte at 3E1F H to 3F H.

ALTERNATE METHOD 2

Boot into NSDOS using the D04B01.NSI floppy-disk image file. Then follow as shown in the session below:
(user input in **bold**)

+**GO HD5XDOS**
North Star Hard Disk Operating System, Version 2.2.0

=**LF HD5XDOS,1 5000**

=**DH 7410-741F**
7410 C1 24 E6 7F FE 40 28 5A FE 5F 28 CD FE 7F 28 C9

=**DS 741B**
741B CD= 43

=**DS 741F**
741F C9= 3F

=**SF HD5XDOS,1 5000**

=

DO SIMILAR FOR THE 'TRANSIENT' PROGRAM ON THE HARD-DISK

Load the TRANSIENT file into RAM at 6F00 H : 'LF TRANSIENT 6F00'.

Alter the bytes required as in Alternate Method 2. They will be in the same locations.
(NOTE: Some versions of the TRANSIENT file will have the positions at 751B H and at 751F H.)

'SF TRANSIENT 6F00' back into its usual place on the disk.

APPENDIX C.

NORTH STAR HARD-DISK DATA FORMAT

DATA LAYOUT ON HARD DISK.

A hard-disk drive is actually a set of spinning disks (or platters). For each platter there are two heads, one above and one below the platter. Thus a hard-disk drive with two platters has four heads, and each head reads and writes on a separate 'surface'. Because all the heads are moved as a single unit from track to track on the platters, the set of tracks being read from is called a 'cylinder', so, in this case, there would be four tracks within each cylinder.

STRUCTURE OF SINGLE TRACK

Each North Star hard disk track consists of 16 sectors. Each sector has its own set of data fields. As the platter spins the disk-drive electronics supply pulses which specify when the first sector of the set of 16 sectors is reached by the read/write head (the index pulse), and when the start of each sector begins(the sector pulse). The index pulse is not retained by the North Star Hard-Disk Controller, but the sector pulse is latched on and is turned off by the Hard-Disk Controller itself.

STRUCTURE OF THE WRITTEN DISK SECTOR

When the sector-pulse is received from the hard-drive by the hard-disk controller, the controller waits a short period then begins sending a stream of zero bytes (00 H). This is to cushion variations in speed of the physical drive. After a enough time has passed, a Sync Byte (01 H) is sent to the hard drive to signify the actual start of the data to write on the disk sector.

The first set of real data written is the Sector-Label Header field, this is a set of nine bytes which identify which sector is being written. This information is later used when reading the disk, to ensure that the data being read is from the sector desired and not another sector.

The next data field contains the 512-bytes of data or program we want to store.

The last data field contains CRC information to ensure that the data has been written cleanly. If the data read back from the disk-sector does not match the store CRC value, there has been corruption of the data.

STRUCTURE OF THE SECTOR-LABEL HEADER FIELD

Example:

PHY	CYL	HED	LSl	LSh	STl	STh	CRC	CRC~
05	0C	83	BD	04	B0	04	09	F6

In typical North Star Computers fashion, the sector ID label is not that as suggested by Shugart in the ST506 protocol, but one which was designed by North Star themselves. However there are similarities.

Byte 1: PHYSICAL SECTOR

The lower 4 bits (Bits 0-3) are used to specify the physical sector on the track. The physical sector is the one calculated by skewing the reads to improve reading/writing speeds. The physical sector is calculated by adding 8 to the ODD logical sectors: logical sector 1 is at physical sector 9, logical sector 15 is at physical sector 7.

Bits 4 and 5 contain the 2-bit overflow of the CYLINDER byte (Byte 2) which then gives the CYLINDER byte a total of 10 bits which allows a maximum of 1024 cylinders

Byte 2: CYLINDER

This byte plus the extra 2 bits specified in Byte 1 allow 1024 cylinders.

Byte 3: Surface (Head Number)

The lower 3 bits are used to specify which head is selected.
The high bit (Bit 7) may used to specify whether the sector is write-protected or not.

Bytes 4 - 5: LOGICAL SECTOR NUMBER

These bytes contain the logical sector-number on the hard-drive. This number may differ from the physical sector number because of the skewing described above.

Bytes 6 - 7: SHIFTED TRACK NUMBER

These bytes contain the logical sector-number on the drive modulo 16. This can be thought of as either the disk-address of sector 0 on the track, or the 12 bits of the track number shifted up 4 bits. This supplies the physical sector address quite simply by adding the PHYSICAL sector in Byte 1 to this up-shifted track number.

example: (In hex numbers as it makes it easier to see.)

```
Logical sector : 04BD H
Track Number   : 004B H
Shifted Track  : 04B0 H
PHYSICAL       :   05 H

Physical Sector: 04B5 H
```

Byte 8: CRC SUM

This byte contains the lower 8 bits of the total obtained by adding all 7 previous bytes.

Byte 9: CRC BYTE COMPLEMENT

This byte contains the complemented CRC byte. (The sum of Byte 8 and Byte 9 is always FF H)

FURTHER EXAMPLE:

PHY	CYL	HED	LSl	LSh	STl	STh	CRC	CRC~
25	52	80	CD	DE	C0	DE	40	BF

Physical Sector: 5 (From Bits 0-3 of PHY)

Cylinder : 52 H (From CYL) + 0200 H (From Bits 4 & 5 of PHY) = 0252 H = 594 (Dec.)

Head : 0 (From Bits 0-2) of HED

Logical Sector : DECD H = 57037 (Dec.)

Physical Sector: 5 (From PHY) + DEC0 H (From Shifted Track) = DEC5 H = 57029 (Dec.)

CRC : 25 H + 52 H + 80 H + CD H + DE H + C0 H + DE H = 440 H = 40H

CRC~ : 40 H complemented = BF H (or BF H + 40 H = FF H)

APPENDIX D.

NSE's HARD-DISK IMAGE FILE STRUCTURE

The hard-disk image structure's size varies according to the number of sectors which were in the original physical hard disk.

The sectors are laid out as in physical sectors, rather than logical sectors. This means the sectors in the disk-image are interleaved, just as they are on the physical disk. There is an unskewing utility in the nse_tools directory, but I don't think this would ever be used by most users of NSE.

NOTE: Validation that the file is truly a North Star Emulator hard-disk image as of NSE, version 0.54 depends solely on the presence of the North Star 'magic' bytes (00 H, FF H) at the start of the first sector of the hard disk image-file. This first sector is North Star's "Hard Disk Label" and contains much information about the size and layout of the hard disk.

If the two validation bytes are not found, NSE will not mount the file at all. While this means that a hard disk image file may become unusable very occasionally, it serves to guard against unwanted accidental damage to other types of files. If warranted, further tests for disk image validity may be included in later versions of NSE.

For producing NSE hard-disk image files of the 'standard' hard disks used by North Star Computers, see under NSE Tools, **mkhd.**