Table of Contents

Introduction

This document describes details of an implementation of Networking on HDOS. The primary implementation is for file access over the network. A secondary implementation (see appendix) is planned for disk images over the network (network jukebox).

Networked Files may be accessed directly by programs that require only .OPEN, .CLOSE, .READ, and .WRITE. This includes many functions of PIP.ABS and some compilers/assemblers/linkers. Note that ASM.ABS probably won't work, as it employs .POSIT (random access) which HDOS does not abstract for non-directory devices. Also, programs or actions that use .DELET, .RENAM, or .LINK will not work on network files. Also, because of the lack of built-in support for filename wildcards, actions the use wildcards to act on many files will not work on network drives.

Copying files to/from network drives is fairly natural, using PIP:

```
PIP SY1:=NW5:MYFILE.EXT
```

It is also expected that some compilers and tools will work. For example:

```
C -M1 NW2:MYPROG.C
```

For other uses, specialized programs are required.

TODO: Add network printer support.

Commands

NETSTAT

Display status of network and mappings. Shows the local node ID, network status byte, and mappings for devices "NW0:" through "NW7:".

NETWORK

Change network mappings. Local device names are "NW0:" through "NW7:". Note that while conventional HDOS device names may be used on the server (e.g. "SY0:" or "DK6:"), that is not required. The device names need only conform to the syntax of two letters followed by one digit. The server must have a mapping of that device name to a directory/folder.

NETDIR

List files on a networked device. The specified device must be a network drive. Wildcards may be used. The default drive will be "NW0:" and the default pattern will be "*.*". The option "/BRIEF" is recognized.

NETDEL

Delete files on a networked device. The specified device must be a network drive. Wildcards may be used, in which case a confirmation will be required before performing the delete.

NETREN

Rename a file on a networked device. The specified device must be a network drive. Wildcards are not allowed.

NETPIP

TBD. This is required to support wildcard file copies, and possibly other features of PIP on network files.

Device Driver

The device driver, "NW", uses the HDOS "non-directory" device flag to achieve a partial file-oriented interface. HDOS does not fully support all features desired for networked files and devices, so the "Device Specific Functions" interface is also used for additional features. The basic driver will

intercept .OPEN, .CLOSE, .READ, and .WRITE system calls on networked files. This does not include .POSIT, .DELET, .RENAM, .CHFLG, or .LINK. Additional festures are added by special programs.

The driver keeps a Network Configuration Table, similar to CP/NET except that it only maps 8 units (drives) and each unit's entry is 4 bytes long. These mappings may be displayed using the NETSTAT.ABS utility and altered using NETWORK.ABS. The first byte of each entry is the remote server node ID, or 0ffh is it is not mapped. The next three bytes contain the device name and unit (in ASCII) on the remote server.

Device Specific Functions

The following functions are added by calling the device driver entry with A=DC.DSF and C having the function number. Generally, this function number is the same as the corresponding HDOS system call. There are also some new functions added to support getting a directory listing.

C=**.CHFLG** D=*bits* E=*mask* HL=*{def-blk, file-string}*
> Change the flags on the file. Not all flags are supported on the server. Write protect should always work. SYS may not if the server does not support additional permissions on files. *{def-blk, file-string}* is an area that contains two addresses, first is the default block and second is the file string.

C=**.POSIT** B=*channel* DE=*sector*
> Set the file pointer to *sector*. The channel must be for an open, networked file.

C=**.RENAM** DE=*def-blk* HL=*{old-file-string, new-file-string}*
> Rename the file. .DECODE will be applied to both *old-file-string* and *new-file-string* and must describe a network device unit.

C=**.DELET** HL=*file-spec*
> Delete file(s). *file-spec* is similar to the results of the .DECODE system call, without the reserved fields, except that it may contain wildcards ('*' must be expanded to '?').

C=**.LINK** DE=*def-blk* HL=*file-string*
> This function is not implemented. It remains to be seen if it can be. On the network side, this consists of .OPENR, .READ, and .CLOSE. HDOS has other requirements for loading and starting a program which may not be accessible/adaptable.

C=**.SERF (11H)** DE=*buffer* HL=*file-spec*
> Search First. CY is set for errors, with A containing the error code (including EC.EOF). On success, *buffer* will be filled with a 23-byte directory entry. This entry will differ from a local HDOS directory entry in that the DIR.FGN and DIR.LGN bytes will contain the size of the file in 256-byte sectors, little-endian (16-bit value). *file-spec* is as specified for .DELET.

C=**.SERN (12H)** DE=*buffer*
> Search Next. Same results as for Search First.

C=**.NTCFG (**45**H)**
> Get Network Configuration Table. Returns HL=address of table.

Network Messages

The network message format is similar to the CP/NET message format, except the format byte 10H is used for requests and 11H for responses. The general message format is:

FMT	DID	SID	FNC	SIZ	payload

Where FMT is the message format as specified by CP/NET, using 10H and 11H for HDOS. DID and SID are the destination and source node IDs, respectively. The response swaps the node IDs compared to the request. FNC is the function to be performed. SIZ is the size of the payload, minus 1. Payload is always at least one byte.

.READ

10H	DID	SID	04H	00H	CHA

11H	DID	SID	04H	00H	EC
11H	DID	SID	04H	FFH	256-bytes

.WRITE

10H	DID	SID	8xH	FFH	256-bytes

11H	DID	SID	8xH	00H	EC

8xH: Channel number with high bit set

.OPENR, .OPENW, .OPENU

10H	DID	SID	FNC	0EH	CHA	file-spec

11H	DID	SID	FNC	00H	EC

.CLOSE

10H	DID	SID	26H	00H	CHA

11H	DID	SID	26H	00H	EC

.POSIT

10H	DID	SID	27H	02H	CHA	sector

11H	DID	SID	27H	00H	EC

.DELET

10H	DID	SID	28H	0DH	file-spec

11H	DID	SID	28H	00H	EC

.RENAM

10H	DID	SID	29H	1BH	old-file-spec	new-file-spec

11H	DID	SID	29H	00H	EC

.CHFLG

10H	DID	SID	30H	0FH	mask	bits	file-spec

11H	DID	SID	30H	00H	EC

.SERF

10H	DID	SID	11H	0EH	vers	file-spec

11H	DID	SID	11H	00H	EC
11H	DID	SID	11H	16H	dir-entry

.SERN

10H	DID	SID	12H	00H	---

11H	DID	SID	12H	00H	EC
11H	DID	SID	12H	16H	dir-entry

Future Work

Network Disk Images

An alternate device drive is planned that allows (up to) 8 disk images to be accessed on a remote server. These images are "raw sector images" such as H8D. Note that accessing files in these images on the server requires special tools to work on the HDOS disk images. The device driver will set the "directory" flag, and implement the DC.REA and DC.WRI commands as raw/direct sector read/write.

New message formats:

.RREAD

10H	DID	SID	34H	04H	device	sector

Device is 3-byte device identifier.

11H	DID	SID	34H	00H	EC	
11H	DID	SID	34H	FFH	256-bytes	

.RWRIT

10H	DID	SID	35H	85H	device	sector	blk	128-bytes

Blk is 0/1 for first/second half of 256-byte sector.

11H	DID	SID	35H	00H	EC