



REMOTE HOST FACILITY HANDBOOK

**CDC® OPERATING SYSTEM:
NOS 1**

Preliminary Edition



REMOTE HOST FACILITY HANDBOOK

**CDC® OPERATING SYSTEM:
NOS 1**

Preliminary Edition

LIST OF EFFECTIVE PAGES

New features, as well as changes, deletions, and additions to information in this manual, are indicated by bars in the margins or by a dot near the page number if the entire page is affected. A bar by the page number indicates pagination rather than content has changed.

PAGE	REV	PAGE	REV	PAGE	REV	PAGE	REV	PAGE	REV
Front Cover	-	5-9	01						
Title Page	-	5-10	01						
ii	01	5-11	01						
iii/iv	01	5-12	01						
v	01	5-13	01						
vi	01	5-14	01						
vii	01	5-15	01						
viii	01	5-16	01						
ix	01	5-17	01						
1-1	01	5-18	01						
1-2	01	5-19	01						
1-3	01	5-20	01						
1-4	01	5-21	01						
2-1	01	5-22	01						
2-2	01	5-23	01						
2-3	01	A-1	01						
2-4	01	A-2	01						
2-5	01	A-3	01						
2-6	01	A-4	01						
2-7	01	A-5	01						
2-8	01	A-6	01						
2-9	01	A-7	01						
2-10	01	A-8	01						
2-11	01	B-1	01						
3-1	01	B-2	01						
3-2	01	B-3	01						
3-3	01	B-4	01						
3-4	01	B-5	01						
3-5	01	B-6	01						
4-1	01	B-7	01						
4-2	01	B-8	01						
4-3	01	B-9	01						
4-4	01	B-10	01						
4-5	01	B-11	01						
4-6	01	B-12	01						
4-7	01	B-13	01						
4-8	01	B-14	01						
4-9	01	Index-1	01						
4-10	01	Index-2	01						
4-11	01	Index-3	01						
4-12	01	Index-4	01						
4-13	01	Index-5	01						
4-14	01	Comment							
4-15	01	Sheet	01						
4-16	01	Back Cover	-						
4-17	01								
4-18	01								
4-19	01								
4-20	01								
4-21	01								
4-22	01								
5-1	01								
5-2	01								
5-3	01								
5-4	01								
5-5	01								
5-6	01								
5-7	01								
5-8	01								

PREFACE

This manual provides reference and operating information for CDC® Remote Host Facility (RHF) Networking. RHF is a software system that links the NOS operating system with other operating systems or with another NOS operating system.

The NOS operating system is used with a CDC CYBER 70/Model 71, 72, 73, or 74 Computer System, a CDC CYBER 170/Model 17X or 7XX Computer System, or a CDC 6000 Series Computer System. The RHF software allows any of the computer systems capable of running NOS to function as an RHF host for either a NOS or non-NOS system computer.

Section 2 is for programmers who will be using RHF to transfer files back and forth between mainframes. Section 3 is for operations personnel in charge of running the mainframes linked by RHF. Section 4 is for the site personnel in charge of installing and maintaining RHF on the NOS systems. Section 5 is for applications programmers using the RHF Access Method. Applications programmers will also find useful information in the NAM 1 Reference Manual and in the NAM 1 FORTRAN Applications Programmer's System Bulletin.

RELATED PUBLICATIONS

Operating information for the CYBER 170 Computer System is not included in this manual; the operating information is found in the NOS Operator's Guide.

<u>Control Data Publication</u>	<u>Publication Number</u>
NOS 1 Reference Manual, Volume 1	60435400
NOS 1 Reference Manual, Volume 2	60445300
NOS 1 Operator's Guide	60453600
NOS 1 Installation Handbook	60435700
NOS 1 Diagnostic Handbook	60455720

NAM 1 Reference Manual	60499500
NAM 1 FORTRAN Applications Programmer's System Bulletin	60480400
380-170 Network Access Device Reference Manual	52653815

DISCLAIMER

This product is intended for use only as described in this document. Control Data cannot be responsible for the proper functioning of undescribed features or parameters.

CONTENTS

1. INTRODUCTION	1-1
System Description	1-1
RHFAM	1-1
RHFAM Activities	1-2
Multimainframe Configurations	1-2
NOS LCN - Minimum Configuration	1-3
Sample Configurations	1-4
Hardware Requirements	1-4
2. PROGRAMMER'S INFORMATION	2-1
MFLINK Control Statement	2-1
MFQUEUE Control Statement	2-7
ST Parameter on the ROUTE Control Statement	2-9
ST Parameter on the Job Control Statement	2-10
FCOPY Control Statement	2-11
3. RHFAM OPERATOR	3-1
RHFAM Operator Commands	3-1
Control Commands	3-1
RHFAM Termination	3-2
DSD Enhancements for Linked File Types	3-2
File Name Table (H) Display	3-2
PURGEALL,S	3-2
FNTLIST	3-3
Active Job Queues (Q) Display	3-3
QTF Operator Commands	3-4
4. INSTALLATION AND MAINTENANCE INFORMATION	4-1
Network Access Device Addresses	4-1
Trunk Control Unit Enable Mask	4-1
Network Access Device EST Entry	4-1
Access Code	4-2
Deadstart NAD Autoload Control	4-3
Deadstart Logical ID Table Setup	4-4
RHFAM Initialization	4-4
Procedure File for RHFAM Subsystem	4-5
RHFAM Startup	4-5

NAD Autoload	4-6
Subsystem Installation	4-7
COMSRHF Parameters	4-7
COMSNAD Parameters	4-8
COMQAPR Parameters	4-8
RHFAM Parameters	4-9
RHF Application Installation Options	4-9
PID/LID File	4-12
RHFAM Dedicated Path Manager Option	4-12
RHFAM Detail Status Logging Option	4-13
NAD Autodump	4-13
Remote Host Facility Operator Utility (RHFOU)	4-14
Network Address Table Display	4-15
Description of the NADT Display	4-15
Network Address Table Command	4-15
Application Table Display	4-16
Description of the APPL Display	4-16
Application Table Commands	4-16
Logical ID Table	4-17
Description of LID Display	4-17
Logical ID Table Command	4-17
Maintenance Logging Transfer Facility (MLTF)	4-18
Accounting Summary	4-18
Accounting Messages	4-19
Input File Staging Accounting	4-20
Output File Staging Accounting	4-20
Permanent File Staging Accounting	4-21
Validation Summary	4-21
Network Failure Processing	4-22

5. RHF ACCESS METHOD INFORMATION 5-1

Facility Interface Program (FIP)	5-1
Module Descriptions	5-1
NETON	5-2
NETOFF	5-4
NETWAIT	5-5
NETGET	5-6
NETGETL	5-7
NETPUT	5-9
NETXFR	5-9
NTPXFR	5-11

NETDBG	5-11
Data Block Header Formats	5-13
Application Block Header Format for Input Data Blocks	5-13
Application Block Header Format for Output Data Blocks	5-16
Supervisory Block Header Formats	5-19
Application Block Header Format for Input Supervisory Messages	5-19
Application Block Header Format for Output Supervisory Messages	5-22

APPENDIXES

A. GLOSSARY	A-1
B. MESSAGES	B-1

INDEX

FIGURES

1-1 MMF Network	1-3
1-2 Interface Functions	1-3
1-3 Multilink NOS Systems	1-4
1-4 Multilink Mixed Systems	1-4
4-1 Sample Configuration	4-11

TABLE

3-1 Additional NOS Control Commands	3-1
-------------------------------------	-----

SYSTEM DESCRIPTION

A Loosely Coupled Network (LCN) is a network of physically connected computer systems. Each computer system is called a mainframe. An RHF/LCN system can include any of the following: CYBER 70/Model 71, 72, 73, or 74 Computer Systems; CDC CYBER 170/Model 17X or 7XX Series Computer Systems; CDC 6000 Series Computer Systems; CDC CYBER 205 Computer System. In addition CDC offers LCN support for IBM OS/MVS JES2 or JES3 as a special product.

The RHF/LCN environment allows jobs, data files, and messages to be transmitted from one computer system to another. Hardware connections between operating systems are established by the Loosely Coupled Network (LCN). The software package that establishes communications between NOS and the LCN is called the Remote Host Facility Access Method (RHFAM). RHFAM serves as the directing executive for all linked operations involving its associated mainframe.

RHFAM

RHFAM is a software package that links the operating system of physically connected computer systems. RHFAM executes at a control point under the NOS operating system.

Each mainframe in a multimainframe system is equipped with an RHF that executes under the operating system of the mainframe. The mainframe at which the RHF is executing is the host mainframe. The mainframe with which the RHF is communicating is a linked mainframe. For example, consider a NOS system computer (SC1) connected to another NOS system computer (SC2) as in figure 1-1. The RHF at SC1 considers SC1 to be the host mainframe and SC2 to be the linked mainframe. At the same time, the RHF at SC2 considers SC2 to be the host mainframe and SC1 to be the linked mainframe.

RHFAM ACTIVITIES

The RHFAM executive performs the following functions:

- . Provides an access method by which application programs on the NOS system can establish logical connections to applications on linked mainframes.
- . Provides multiplex data flow.
- . Provides the ability to auto-start an application that has been requested by a linked mainframe.

MULTIMAINFRAME CONFIGURATIONS

A mainframe is identified by a host Physical ID (PID) that consists of three display code characters (letters or digits). In a multiframe (MMF) system, the last two characters of each PID is the NOS MMF ID and must be unique. The NOS PID is formed by prefixing the MID with an M. The host PID of a mainframe is called the physical or linked ID of that mainframe at the linked mainframes.

For example, a physical ID at a mainframe, mainframe AA, may be MAA. At another mainframe, mainframe BB, the physical ID may be MBB. To mainframe AA, AA is the host MMF ID and MBB is the linked ID. However, to mainframe BB, BB is the host MMF ID and MAA is the linked ID.

NOS LCN - MINIMUM CONFIGURATION

Two CYBER 170 Computer Systems can be connected to each other via the LCN. Figure 1-1 illustrates a minimum configuration, and Figure 1-2 illustrates interface functions.

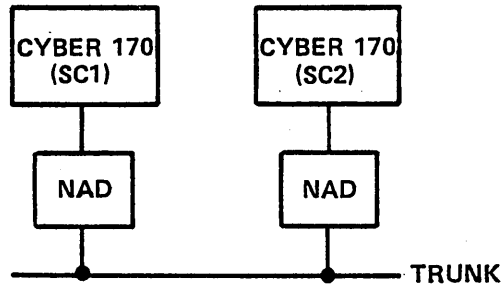


Figure 1-1. MMF Network

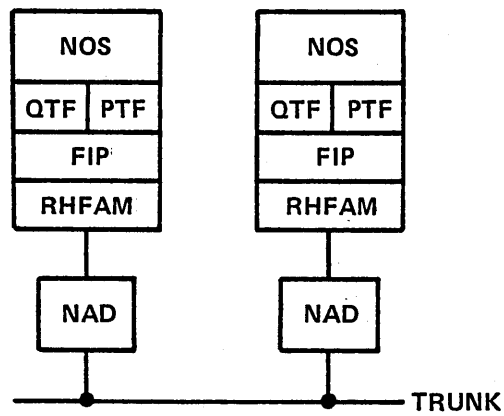


Figure 1-2. Interface Functions

SAMPLE CONFIGURATIONS

Two or more mainframes can be connected to each other via the LCN. Figures 1-3 and 1-4 illustrate typical LCN configurations.

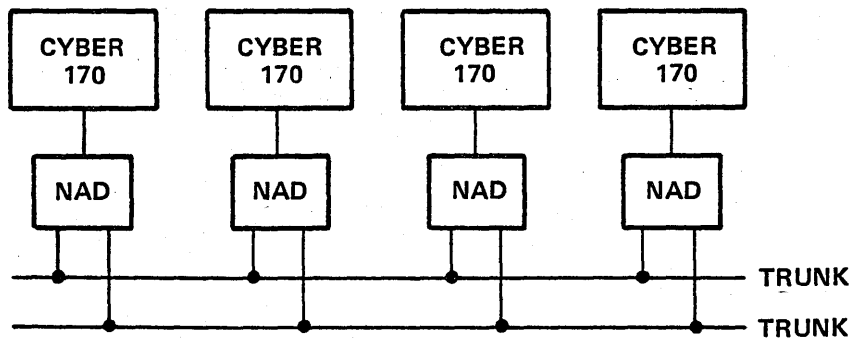


Figure 1-3. Multilink NOS Systems

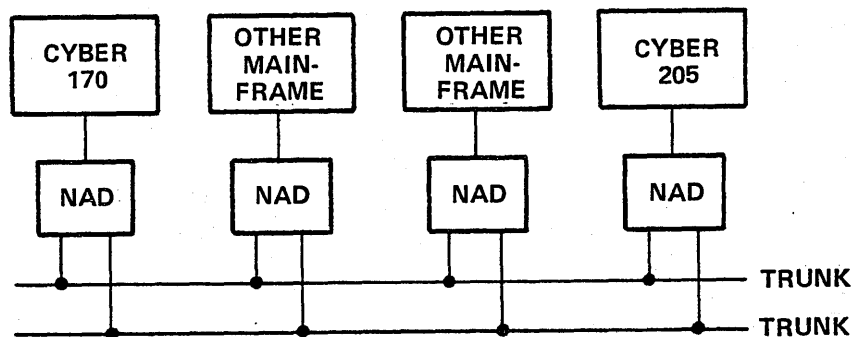


Figure 1-4. Multilink Mixed Systems

HARDWARE REQUIREMENTS

The minimum hardware requirement for NOS with RHFAM is the same as the minimum for NOS with the addition of the following:

- One Network Access Device (NAD) (65 K bytes)
- One Trunk Control Unit and its associated Data Set

The maximum LCN configuration on a single NOS system is as follows:

- Four Network Access Devices (NAD) (131 K bytes)
- Four Trunk Control Units on each NAD and their associated Data Sets

MFLINK CONTROL STATEMENT

The MFLINK control statement allows the user to send a text file from the host mainframe to a linked mainframe to accomplish the following:

- File transfers between the host mainframe and the linked mainframe. The user can store files on and retrieve files from the linked mainframe.
- File alteration and maintenance on the linked mainframe. The user can change the size of a file, change the passwords associated with a file, purge a file, or whatever is allowed by the Permanent File Transfer Facility (PTF) on the linked mainframe.

The text file consists of lines of text that resemble control statements on the linked mainframe. The PTF on the linked mainframe determines the syntax of the lines of text. The PTF on the host mainframe does not process the text file, but rather sends it unaltered to the linked mainframe. The PTF on the linked mainframe processes the text file as documented in the reference manual for the RHF on that mainframe. The user determines which lines of text will appear in the text file.

The format of the MFLINK control statement is:

MFLINK (lfn,ST=lid,I=textfile,DD=dd,EP,RT)

If lfn is specified, it must be the first parameter; the others are order-independent.

- lfn Optional local file name to be used in any file transfers. The local file must be disk resident. If the transfer is from the host mainframe to the linked mainframe, lfn must be local to the job before MFLINK is called. If the transfer is from the linked mainframe to the host mainframe, MFLINK either writes over the existing lfn, or if there is no lfn local to the job, MFLINK creates a new file with the name lfn. If lfn is omitted, but subsequent lines of text call for a file transfer, MFLINK uses LFILE as the default lfn. Some lines of text, such as CHANGE, CHARGE, and PACKNAM (for the NOS PTF) do not require an lfn.
- ST=lid Specifies the logical identifier of the mainframe to which PTF is to send the text file. The ST parameter must be specified on the first and only the first MFLINK statement of a series (session) that are for the same mainframe. The occurrence of the ST parameter on the MFLINK statement initiates a new MFLINK session with the linked mainframe. There is no carry-over from one session to the next. No other control statements can appear in an MFLINK session. The appearance of a control statement other than MFLINK terminates the session.
- I=textfile Specifies the file containing the lines of text that PTF is to send to the linked mainframe. The default textfile is INPUT.
- DD=dd Specifies the file conversion mode for file transfers. The available conversion modes are:
- | | |
|----|--|
| US | Undefined, structured (default) |
| UU | Undefined, unstructured |
| C6 | 6-bit character representation, structured |
| C8 | 8-bit character representation, structured |

- EP Specifies the error processing RHF is to do if network problems cause a loss of the link during a file transfer. If EP is omitted, the PTF on the linked mainframe returns the file and the PTF on the host mainframe retries the request. Specifying the EP parameter inhibits the retry.
- RT Specifies real-time action RHF is to take in response to subsystem errors such as SUBSYSTEM NOT ACTIVE or LID DISABLED. If RT is specified, PTF aborts on detecting a subsystem error. If RT is omitted, after an installation defined period of time PTF retries the request that was being processed when the error was detected.

The lines of text processed by the NOS PTF resemble a subset of the NOS control statements. With a few exceptions the lines of text operate in the same manner as their NOS control statement counterparts. ATTACH and DEFINE operate essentially the same as GET and SAVE: that is, ATTACH retrieves a copy of the direct access permanent file and DEFINE copies the local file onto a direct access permanent file. The NA option works somewhat differently in interactive mode through MFLINK than it does in normal interactive mode. If the NA option is omitted and any error condition occurs, or if the NA option is specified and an error condition that is not temporary occurs, PTF returns an error message to the terminal and waits for further lines of text. If NA is specified and a temporary error occurs, PTF waits for the error condition to end and then processes the line of text. During this waiting period MFLINK will terminate (time-out) if an installation defined period of time elapses with no activity. The release value for this period is ten minutes. The lines of text do not all have the same syntax as their NOS control statement counterparts. The differences are documented here. Refer to the NOS 1 Reference Manual, Volume 1, for explanation of the parameters. Only the parameters listed here are supported.

The USER and CHARGE (if required) must be the first two lines of text.

USER(usernum,passwd,familyname)

Operates the same as its NOS control statement counterpart.

CHARGE(chargenum,projectnum)

Operates the same as its NOS control statement counterpart.

APPEND(pfn/UN=username,PW=password,PN=packnam,R=r,NA)

Adds the file specified on the MFLINK statement to the end of pfn.

ATTACH(pfn/UN=username,PW=password,PN=packnam,R=r,N,RT)

Transfers a copy of pfn to the host mainframe and gives it the name specified on the MFLINK statement. No interlock is maintained between lfn and pfn, except during the file transfer (as if M=READ had been specified).

CHANGE(nfn=ofn/PW=password,CT=ct,M=m,BR=br,PR=pr,PN=packnam,R=r,NA,CE)

Operates the same as its NOS control statement counterpart.

DEFINE(pfn/PW=password,CT=ct,m=m,BR=br,PR=pr,PN=packnam,R=r,S=space,NA)

Creates a direct access file of name pfn and copies onto it the file specified by lfn on the MFLINK control statement. There is no provision to copy over an existing direct access file; the existing file must first be purged. No interlock is maintained between lfn and pfn, except during the file transfer.

GET(pfn/UN=username,PW=password,PN=packnam,R=r,NA)

Transfers a copy of pfn to the host mainframe and gives it the name specified on the MFLINK statement.

PACKNAM(PN=packnam)

or

PACKNAM(packnam)

Operates the same as its NOS control statement counterpart.

NOTE

It is possible to create a deadlock condition between two mainframes, if they both issue a PACKNAM for the same pack.

PERMIT(pfn, usernum=m/PN=packnam, R=r, NA)

Operates the same as its NOS control statement counterpart.

PURGE(pfn/UN=usernum, PW=passwd, PN=packnam, R=r, NA)

Operates the same as its NOS control statement counterpart.

REPLACE(pfn/UN=usernum, PW=passwd, PN=packnam, R=r, NA)

Replaces pfn with the lfn specified on the MFLINK control statement.

SAVE(pfn/PW=passwd, CT=ct, M=m, PN=packnam, R=r, NA)

Creates an indirect access file of name pfn and copies onto it the file specified by lfn on the MFLINK control statement.

NOTE

All requests support only one pfn per text line.

Example using MFLINK control statement:

```
JBLINK(T77)
USER(XYZ,UVW,RST)
CHARGE(CHAR24,PROJ24)
GET(NEWFILE/PW=JUICE)
MFLINK(MYFILE,ST=AAA)      ①
MFLINK(NEWFILE)           ①
7/8/9
USER(ABC,DEF,GHI)         }
CHARGE(CHAR16,PROJ21)     } ②
GET(OLDFILE/UN=CBA,NA)    }
7/8/9
SAVE(NUFILE/PW=DAVE,NA)   } ③
PURGE(ZEEFILE/UN=DBM,PW=SECRET,NA) }
6/7/8/9
```

- ① Textfile will be read from INPUT by default.
- ② Textfile for first MFLINK statement. It establishes the user's validation on mainframe AAA and transfers a copy of OLDFILE back to the host mainframe using the name MYFILE.
- ③ Textfile for second MFLINK statement. It transfers a copy of NEWFILE to AAA and saves it there as NUFILE. It also purges ZEEFILE from AAA.

MFQUEUE CONTROL STATEMENT

The MFQUEUE control statement allows the user to send a local file to the input or output queue of a linked mainframe. The MFQUEUE control statement functions much like the NOS ROUTE control statement, especially if the linked mainframe is also running NOS. The format of the MFQUEUE control statement follows

MFQUEUE(lfn,ST=lid,DC=dc,DD=dd,I=textfile)

If lfn is specified, it must be the first parameter; the others are order-independent.

lfn Name of the local file to be routed to the linked mainframe. Default is LFILE.

ST=lid Specifies the logical or physical identifier of the linked mainframe to which the file is to be routed. The physical identifier of the host mainframe is the default.

DC=dc Disposition code as defined for the ROUTE control statement in the NOS 1 Reference Manual, Volume 1. Specifies the disposition code for the file on the host mainframe, and, if not overridden by the routing text, the disposition code for the file on the linked mainframe. The Queue File Transfer Facility (QTF) places the file into a special queue on the host mainframe with an attribute that reflects the specified disposition code. Depending on the ST parameter QTF then processes the file on the host mainframe or sends it to the linked mainframe. The default disposition code is PR. Special file names, such as OUTPUT and PUNCH have no effect on the disposition of the file. DC=IX can be specified as explained later for the DC parameter in the textfile.

DD=dd Specifies the file conversion mode for file transfers. The available conversion modes are:

US	Undefined, structured (default)
UU	Undefined, unstructured
C6	6-bit character representation, structured
C8	8-bit character representation, structured

I=textfile The user can send a line of explicit routing text with the file to the linked mainframe. The QTF on the linked mainframe processes the file according to the routing text. I specifies the file that contains this routing text. The default for I is INPUT. The file can contain one line of routing text up to 150 characters long. The line of routing text accepted by the NOS QTF resembles closely the NOS ROUTE control statement. The differences are documented here. Only the parameters listed here are supported. The format of the routing text accepted by the NOS QTF follows.

ROUTE(P₁,P₂,...,P_n)

The parameters p_i can be DC,EC,FC,IC,ID,REP,SC,FM,TID, and UN in the forms documented for the ROUTE control statement. Note that the lfn parameter required on the ROUTE control statement is not supported for the MFQUEUE routing text; the user specifies the lfn in the MFQUEUE control statement.

UN=xx If the user chooses to send routing text with the file, he must include the UN parameter in the text. xx can be any user validated on the linked mainframe system.

DC=dc The value specified for dc in the routing text overrides the value specified for dc in the MFQUEUE control statement.

QTF provides a new value for dc. Normally, output files from a job that was sent to the input queue on the linked mainframe via MFQUEUE return to the output queues on the host mainframe from which the job was sent. Some mainframes, such as the CYBER 200 series, do not have input/output devices connected to them. Thus QTF provides the DC=IX disposition code. DC=IX causes the file specified in the MFQUEUE statement to be sent to the input queue of the linked mainframe. Any resultant output files are processed on the linked mainframe.

QTF does not support DC=SC. Special file names, such as OUTPUT and PUNCH have no effect on the disposition of the file.

Refer to the ROUTE control statement in the NOS 1 Reference Manual, Volume 1, for descriptions of the remaining parameters for the routing text.

Example using MFQUEUE control statement:

```
JBCARDS(T77)
USER(XYZ,UVW,RST)           This job transfers local file
CHARGE(CHAR32,PROJ61)      CARDS to the punch queue on
GET(CARDS)                  mainframe XR7.
MFQUEUE(CARDS,ST=XR7)
7/8/9
ROUTE(UN=MYUNUM,DC=PU,EC=026) Textfile for MFQUEUE statement.
6/7/8/9
```

ST PARAMETER ON THE ROUTE CONTROL STATEMENT

The user can specify the ST parameter on the NOS ROUTE control statement to send a file to a queue on a linked mainframe. NOS prepares the designated file as specified by the other parameters, and then the host QTF sends the file to the QTF on the linked mainframe.

The ST parameter is order-independent. The format of the ST parameter for the ROUTE control statement follows.

```
ST=lid
or
ST=pid
```

The ST parameter can specify either the logical identifier or the physical identifier of the linked mainframe.

NOTE

If the ROUTE statement sends a job file to an input queue, the ST parameter on the ROUTE statement overrides any ST parameter that might appear on the job statement in the job file.

Example of the ROUTE statement with the ST parameter:

```
ROUTE(XYZFILE,DC=PR,ST=ABC)
```

ST PARAMETER ON THE JOB CONTROL STATEMENT

If the linked mainframe to which the user wishes to send his job is running NOS, the user can include the ST parameter on the job control statement. The QTF on the host mainframe then sends the entire job, from the job statement through end-of-file, to the QTF on the linked mainframe. The QTF on the linked mainframe puts the job into the input queue. The ST parameter can also be used in the same manner if the linked mainframe is running an operating system that uses a job statement similar to the NOS job statement.

The ST parameter can specify either the logical identifier or the physical identifier of the linked mainframe. It can be used in either the order-independent (keyword) or the order-dependent (positional) form of the job statement. The keyword format of the ST parameter for the job statement follows.

STlid
or
STpid

The positional format of the ST parameter is just lid or pid. It must appear in the fifth parameter position of the job statement.

The following job statements are equivalent.

JBSHOW(CM10000,STABC,T100)

JBSHOW(,100,10000,,ABC)

FCOPY CONTROL STATEMENT

The FCOPY control statement converts a file from one code set to another.

The control statement format is:

FCOPY(P=lf_{n1},N=lf_{n2},PC=cs₁,NC=cs₂,R)

P=lf_{n1} File to be converted (default is OLD). The user should assign lf_{n1} to the job before performing the FCOPY operation.

N=lf_{n2} File on which the converted data from lf_{n1} is written (default is NEW). If lf_{n2} is not assigned to the job, FCOPY creates it.

PC=cs₁ Code set of lf_{n1}(default is ASCII)

NC=cs₂ Code set of lf_{n2}(default is ASCII8)

R If R is specified, lf_{n1} and lf_{n2} are rewound before and after the conversion. If R is omitted, lf_{n1} and lf_{n2} are not rewound before or after the conversion.

The following options are available for PC or NC:

DIS 6-bit display code.

ASCII 6/12 display code

ASCII8 ASCII 8-bit character in 12 bit byte with zero byte line termination.

ASC8 ASCII 8-bit characters in 12 bit byte with ASCII unit separator termination.

RHFAM OPERATOR COMMANDS

The display console of the host mainframe serves as the operating console of the host mainframe and RHFAM.

The operator can issue commands to the host system, or to RHFAM.

CONTROL COMMANDS

The host mainframe operator using the display console has available all standard NOS displays and commands described in the NOS Operator's Guide. Additional available commands are shown in Table 3-1.

TABLE 3-1. ADDITIONAL NOS CONTROL COMMANDS

Command	Action Initiated
ENABLE,RHFAM.	Enable starting of RHF with AUTO command.
DISABLE,RHFAM.	Disable starting of RHF with AUTO command.
n.RHFffff.	Initiate RHFAM at control point n.

RHFAM TERMINATION

RHFAM termination is the logical disconnection of RHFAM from all the linked mainframes. All pending requests to the linked mainframes are dropped when RHFAM is dropped.

If RHFAM is connected to one linked mainframe, RHFAM can be dropped to disconnect the link. If RHFAM is connected to more than one linked mainframe, the drop disconnects all linked mainframes. The operator can gracefully terminate RHFAM activity at the linked mainframes one at a time by the use of the Disable Logical ID command.

The operator should perform the following to idle the RHFAM control point:

Enter the DSD command n.IDLE or the Remote Host Facility Operator Utility (RHFOU) command IDLE to complete all active file transfers and prevent any new activity from being initiated. When all activity completes, the subsystem will terminate.

The operator can also drop RHFAM's control point by entering the command:

n.STOP.

n Control point assigned to RHFAM

DSD ENHANCEMENTS FOR LINKED FILE TYPES

FILE NAME TABLE (H) DISPLAY

The H display can be set to indicate only files of a linked type. For example:

H,S. S = Linked Files.

NO.	NAME	CP	TY	EQ	PR	ID	STAT
11.	ABCAABBB.	.	S1.	1.	1007.	65.	LID

Linked files will have the "LID" displayed in the "STAT" field.

PURGEALL,S.

Purges all linked files from the system.

FNTLIST

A three character logical ID may be specified as a selection criterion.

FNTLIST(LD=ABC)

ABC is the logical ID for selection of remote queue files. If LD=* is specified, all remote queue files are selected.

ACTIVE JOB QUEUES (Q) DISPLAY

The status field of the Q display has the following new status types:

- ND - Waiting for NAD message.
- RH - Waiting for RHFAM.

QTF OPERATOR COMMANDS

QTF offers a left screen K display of current activities as shown here.

QUEUE FILE TRANSFER FACILITY

FILE TYPE TRANSFERS ALLOWED.

INPUT/Y PRINT/Y PUNCH/N SPECIAL/Y

MAXIMUM CONNECTIONS ALLOWED 8

CURRENT NUMBER ALLOWED 4

FILE NAME	LID	TYPE	FILE NAME	LID	TYPE
*APQT039	TS1	IN	DGTEY67	SC2	SP
RTTYQSL	MBB	PR	*HJM3911	GLQ	IN
FLAVIUS	MAA	SP			

TOTAL FILES TRANSFERED 9

LIDS CURRENTLY DISABLED IN QTF
lid₁=iops lid₂=iops lid₃=iops

command entered

The files listed here are awaiting transfer to a linked mainframe. An asterisk preceding a file name indicates that the connection to the linked mainframe has been made, but the transfer has not yet been completed.

QTF can enable/disable individual LIDs for file transfers by file type. The disabled LIDs are listed on the line below the header with an indication of for which of the four file types the LID is disabled.

I INPUT
O PRINT
P PUNCH
S SPECIAL

If there are no disabled LIDs, the header does not appear.

If there is an error in the command entered, the word ERROR appears above the command.

The operator can use the following commands to control the actions of QTF.

AC=x.	Change the current number of connections allowed to x
IDLE.	Idle after all transfers are complete
STOP.	Drop immediately
ON./OFF.	Allow/disallow file transfers for all types of files
ON=xx./OFF=xx.	Allow/disallow file transfers for the specified file type

<u>xx</u>	<u>File Type</u>
IN	Input
PR	Printer
PU	Punch
SP	Special

RESET. Update internal tables to reflect the current state of LIDs as to whether they are linked, enabled/disabled, etc. Also, clears the locally disabled lids as displayed on the bottom of the left screen K Display

These commands are also shown on the right screen of the K display.

NETWORK ACCESS DEVICE ADDRESSES

Each NAD has an 8-bit physical network address. This address is selected by two hexadecimal thumbwheel switches on each NAD. The NAD is normally given a unique address in the network, and each NAD must be unique on any given set of trunks relative to each NAD. The network address may be duplicated if the duplicated address is on a separate trunk and the duplicated address is not known to any NAD which is directly coupled. NOS only supports network addresses in the range 01₁₆ to 7F₁₆.

TRUNK CONTROL UNIT ENABLE MASK

Each NAD may be configured with up to 4 data trunks. The trunks are referenced by a Trunk Control Unit (TCU) enable mask. This four bit mask specifies the TCUs as: 2**0 = TCU3, 2**1 = TCU2, 2**2 = TCU1, 2**3 = TCU0. Although it is not necessary, each trunk is normally connected to the same TCU at each NAD. This will identify the trunk in a consistent manner by all NADs.

NETWORK ACCESS DEVICE EST ENTRY

Standard equipment status table (EST) entries describe the Network Access Device as follows:

NC Network Access Device

At deadstart time, the NAD is defined in the EST as follows:

EQnm NC,ON,0,aa,cc.

nm EST ordinal

aa TCU enables
TCUs are specified by bit 0 = TCU3, bit 1 = TCU2,
bit 2 = TCU1, bit 3 = TCU0.

cc Channel number

This EST entry will cause a default load of NAD controlware at deadstart time.

ACCESS CODE

Each NAD has a 16-bit physical access code. This code is set by 4 hexadecimal thumbwheel switches on each TCU. All messages transmitted by a NAD contain this access code. The TCU in a NAD which receives a message with an invalid access code will not reply to the message and the message is discarded.

The access code is treated as four 4-bit entities. If all 4 bits are zero it is interpreted as a "don't care". For each 4-bit field in the NAD's access code which is zero, the NAD will accept a message with anything in this field.

NAD access code	FOFO
Message access code	<u>01F3</u>
Result	<u>NYYY</u>
	(Y = good compare)
	(N = bad compare)

The access code can be disabled by using an all zero access code on each NAD.

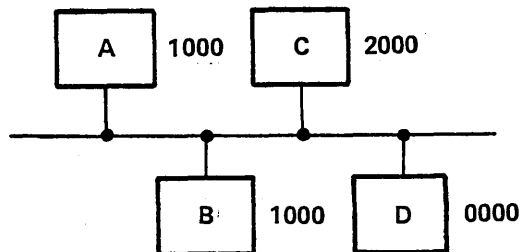
When the NAD is transmitting a message it combines the original message access code in the control message or from the Path Control Table (PCT) with the physical access code to generate the transmitted message access code. Whenever there is a zero field in the physical access code, the original message access code field will be used.

NAD access code	FOFO
Original message access code	<u>01F3</u>
Transmitted message access code	<u>F1F3</u>

A NAD which has no zero fields in its physical access code can generate and will accept a single access code, the one in the NAD thumbwheel switches.

A NAD which has one or more zero fields in its physical access code can generate and will accept a set of access codes.

The following example illustrates the use of the access code:



A, B, and C can communicate with D. A can also communicate with B. No other combinations are possible.

DEADSTART NAD AUTOLOAD CONTROL

This CMRDECK entry identifies the type of controlware to be loaded on specified NAD channels. Controllers may or may not be loaded at deadstart by this entry, depending on specified parameters.

LBC,type,ch1,ch2,...,chn.

<u>Type</u>	<u>Controlware</u>
ND	Load MG101 NAD controlware
NN	Identify channel(s) as having a NAD, but do not load controlware

DEADSTART LOGICAL ID TABLE SETUP

The length of the Logical ID Table may be specified via the CMRDECK as follows:

LIDT=N. Set number of allowed logical IDs to N, where N has a maximum value of 63 and a minimum value of 2. The default is 8.

NOTE - This value must be set to a value at least two larger than the number of LID entries in the IPRDECK.

All logical IDs (LID) which are unique must be described in the IPRDECK at deadstart time. The LID is defined as follows:

LID=AAA,X.

AAA Three alphanumeric characters.

X Attribute for the LID.
H The LID specifies this host.
L The LID specifies a remote host.
D The LID is disabled.

RHFAM INITIALIZATION

Deadstart can toggle the automatic initialization of the RHFAM subsystem with the following IPRDECK entry:

RHFAM.

RHFAM is initialized when the central processor program RHFAM is brought to a control point for execution. RHFAM is initiated by the command:

n.RHFffff

n Control point assigned to RHFAM

RHFffff Calls a procedure file named RHFffff under the system user index (377777B) which initiates the Remote Host Facility (RHF) subsystem.

PROCEDURE FILE FOR RHFAM SUBSYSTEM

The following is the default procedure file which must be installed either on the system file or as a permanent file under index 377777B.

```
RHF
RHFAM.
EXIT.
DMP.
DMP(60000)
IF(EF.EQ.ODE)RETURN(OUTPUT)
```

RHFAM STARTUP

RHFAM is assembled with an installation defined Network Address Table (NADT). This table can be redefined at execution time by creating an indirect access file under the system user index (377777B). The file must be named NADTxx where xx = mainframe id.

The format of the NADT entries is:

LID, EQ, ND, RT, LT, DD, AC, ST.

Where:

LID Logical id of entry
EQ EST ordinal of local NAD
ND Destination NAD address
RT Remote NAD trunk enables
LT Local NAD trunk enables
DD Device address of remote NAD
AC Access code
ST Not used if entry enabled

NOTE: If no NADTxx file is found the installation assembled table will be used and the message "DEFAULT NETWORK ADDRESS TABLE USED." issued.

After RHFAM is initialized, it searches the EST for up to four NAD devices and automatically queries for requests to establish communications. The Queue File Transfer Facility (QTF) initiator application will be auto-started at this time if it is enabled in the RHFAM application table.

NAD AUTOLOAD

The Network Access Devices which are described in the EST as type NC will be autoloaded at deadstart time by default. If it is not desired to have this load take place at deadstart time the operator can use the LBC command with the NN parameter to inhibit loading. The NADs can also be loaded after the system is operational via the LOADBC utility.

The LOADBC control statement provides the capability to dynamically download controlware to the associated NAD. The calling job must be system origin and the system must be in engineering mode (refer to the NOS Operator's Guide).

LOADBC reads the correct controlware from the system file into its field length and calls the PP program 1LC/2LC to download the controlware. 1LC/2LC reads the controlware from central memory while downloading to the proper NAD. LOADBC will issue appropriate messages to indicate the success or failure of the controlware load attempt.

The format of the control statement when loading a local NAD is:

LOADBC,C=cc.

cc Channel of the local NAD.

The format of the control statement when loading a remote NAD is:

LOADBC,C=aa,ND=bb,TY=ccc,LT=dd,AC=eeee,MS=ff.

aa Channel of the local NAD. (This NAD must have been previously loaded).

bb Hexadecimal remote NAD number.(required)

ccc Type of controlware to load (required).

NAD - CYBER 170
IBM - IBM
MIN - Common Minicomputer.

dd Local trunk enable mask (required).

eeee Hexadecimal access code (optional, default is 0000).

ff Memory size (16, 32, 49, or 65; required).

SUBSYSTEM INSTALLATION

Installation parameters for the Remote Host Facility can be modified using the following procedures. Obtain listings of the appropriate decks in order to obtain material, such as line numbers, which is needed when writing code to change installation parameters.

COMSRHF PARAMETERS

COMSRHF contains parameters used for control of RHFAM functions. Assemble RHFAM to obtain a listing of COMSRHF. Parameters for the Network Address Table can be specified via the CONF macro.

<u>Parameter</u>	<u>Default</u>	<u>Comment</u>
MPFI	10B	Maximum permanent file initiators.
MPFS	10B	Maximum permanent file servicers.
MQFS	10B	Maximum queue file servicers.
NATL	50D	Maximum Network Address Table size.
PSWD	PASSWRD	This is a micro definition that specifies the LCN network password.

The CONF macro is used to specify the default Network Address Table entries as follows:

```
LID    CONF    EST,NAD,RT,LT,D,AC,S
```

LID = Logical ID. Three alphanumeric characters.

EST = Local EST ordinal. Octal constant.

NAD = Remote Network Access Device Number. Hexadecimal Constant.

RT = Remote Trunk Control Unit enables. Octal constant.

LT = Local Trunk Control Unit enables. Octal constant.

D = Destination Device address. Hexadecimal constant.

AC = Access Code. Hexadecimal constant.

S = State. Non-null = disabled.

The released Network Address Table is as follows:

```
TS1    CONF    40,7,10,10,0,FOFO
TS1    CONF    41,6,10,10,0,FOFO
TS1    CONF    40,6,10,10,0,FOFO,OFF
TS1    CONF    41,7,10,10,0,FOFO,OFF
M20    CONF    40,F,10,10,0,FOFO,OFF
```

COMSNAD PARAMETERS

COMSNAD contains the Network Access Device (NAD) controlware initialization parameters. The function of each parameter is described in the 380-170 NAD Reference Manual. The default parameters defined in the NADIP macro definition are typical values chosen for the default configuration. If these values are changed after installation, the NAD controlware loader (ILC) must be re-assembled.

Default NAD initialization parameters for a 32K byte NAD:

NAD memory size - 1	7FFF ₁₆
TCU enables	F ₁₆
Stream mode	0
Maximum number of NADs	10
Maximum number of paths	35
Internal buffer count	36
Control message buffer count	36
Type 0 buffer size/count	81 ₁₆ /0
Type 1 buffer size/count	810 ₁₆ /6
Type 2 buffer size/count	0/0
Type 3 buffer size/count	0/0
Trunk receive queue limit	4
Trunk send queue limit	1
Control message receive queue limit	30
Control message send queue limit	30
Path receive queue limit	2
Path send queue limit	1
Trace enables	2954 ₁₆
Character set mode (64)	0

COMQAPR PARAMETERS

COMQAPR contains the RHF applications installation parameters. To obtain a listing of this deck, assemble deck INSTLQ from the RHF program library.

The default values are:

ACNMAXC	=	8	Maximum QTF connections
DELTIM	=	20 sec	Queue access delay for QTF
LIDFRS	=	10 min	LID table refresh time for QTF
MAXRTRY	=	10	Error retry limit for PTF/QTF
NAKRTRY	=	10	Network congested retry limit
NTLMAX	=	203 words	Maximum block size
TERT	=	100 sec	PTFS system event wait time
TIMEOUT	=	600 sec	Application required response time

RHFAM PARAMETERS

Assemble RHFAM to obtain a listing. Parameters for the Application Table can be specified via the APPL macro. The RHFAM listing contains an explanation of the macros.

The released Application Table is as follows:

QTF	APPL		Queue transfer facility initiator
	DUP	MQFS,1	
QTFS	APPL	A	Queue transfer facility servicer
	DUP	MPFI,1	
PTFU	APPL		PF transfer facility initiators
	DUP	MPFS,1	
PTFS	APPL	A	PF transfer facility servicers

RHF APPLICATION INSTALLATION OPTIONS

Two installation selectable options are available to control FIP accountability dayfile message logging and echo checking. These options are normally selected during the build. To deselect either or both options, obtain a deck listing of the RHF installation job from the DECKOPL. This job can then be modified by running the DECKFIX installation job.

To deselect the FIP accountability message logging, remove the *DEFINE ACCNT Modify directive.

To deselect the application echo checking, remove the *DEFINE ECHO Modify directive.

An additional option is available to build FIP with debugging dayfile message capability (refer to NETDBG in section 5). To select this option, insert a *DEFINE FIPDB modify directive into the build procedure using the same format as with ACCNT and ECHO.

Configuration example:

The Loosely Coupled Network (LCN) must be described in both a local and remote perspective. The local definition is made via the Network Address Table in RHFAM's field length. All legal logical IDs (LID) must be defined in the IPRDECK.

Figure 4-1 is an example of a valid configuration. Entries into the CMRDECK, IPRDECK, and the Network Address Table for each mainframe are as follows:

Mainframe MAA

CMRDECK

EQ40=NC,ON,0,14,6.

EQ41=NC,ON,0,14,7.

IPRDECK

LID=MBB,L.

LID=MCC,L.

NETWORK ADDRESS TABLE

MBB CONF 40,3,10,4,0,1234

MBB CONF 40,4,10,4,0,1234

MBB CONF 40,5,10,4,0,1234

MBB CONF 41,3,10,4,0,1234

MBB CONF 41,4,10,4,0,1234

MBB CONF 41,5,10,4,0,1234

MCC CONF 40,6,6,14,0,1234

MCC CONF 41,6,6,14,0,1234

Mainframe MBB

CMRDECK

EQ30=NC,ON,0,14,23.

EQ31=NC,ON,0,14,24.

EQ32=NC,ON,0,14,25.

IPRDECK

LID=MAA,L.

LID=MCC,L.

NETWORK ADDRESS TABLE

MAA CONF 30,1,4,10,0,1234

MAA CONF 30,2,4,10,0,1234

MCC CONF 30,6,14,14,0,1234

MAA CONF 31,1,4,10,0,1234

MAA CONF 31,2,4,10,0,1234

MCC CONF 31,6,14,14,0,1234

MAA CONF 32,1,4,10,0,1234

MAA CONF 32,2,4,10,0,1234

MCC CONF 32,6,14,14,0,1234

Mainframe MCC

CMRDECK

EQ50=NC,ON,0,16,2.

IPRDECK

LID=MAA,L.

LID=MBB,L.

NETWORK ADDRESS TABLE

MAA	CONF	50,1,14,6,0,1234
MAA	CONF	50,2,14,6,0,1234
MBB	CONF	50,3,14,14,0,1234
MBB	CONF	50,4,14,14,0,1234
MBB	CONF	50,5,14,14,0,1234

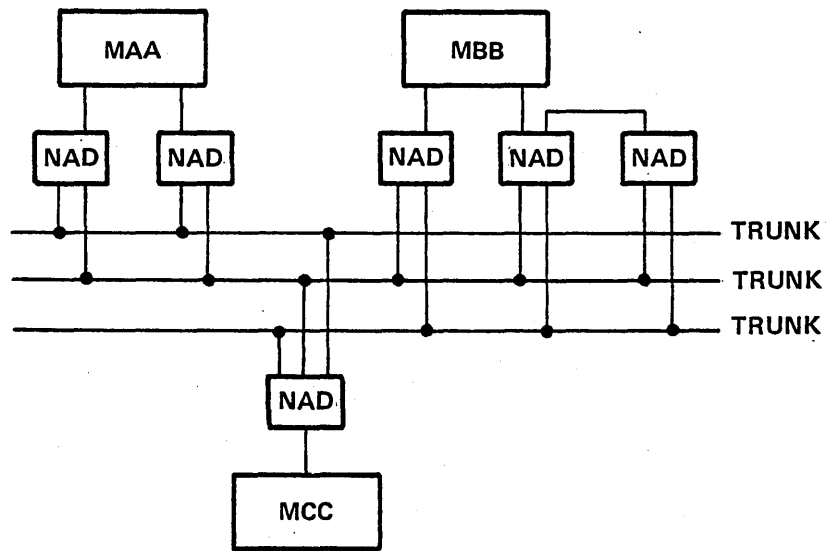


Figure 4-1. Sample Configuration

NOTE: Trunk numbers are 0-2, left to right at the NAD connection. Of the three hosts, none are the same trunk number for the different hosts.

PID/LID FILE

The name of the direct access file that contains LID/PID association is QTFTxx (where xx is the ID of the mainframe that this facility is executing on). This file is required and resides under the 37777B user index. Each entry has the following format.

PID,PID,LD1,LD2,,,LDN.

where PID is the PID of the host that also has LD1 through LDN.
Example:

PAA,PAA,LAB,LAC,LAD,,,.
PAA,PAA,LAJ,,,.
PBA,PBA.
etc.

where the IDs beginning with a P are PIDs and the rest LIDs.
Each line entry can be up to 150 characters long but must have at least two IDs on it.

The maximum number of unique LIDs is 64, and the maximum number of PIDs that can be associated with each LID is also 64. If either is greater or if the IDs are not exactly three alphanumeric characters, the facility issues an error message and aborts.

RHFAM DEDICATED PATH MANAGER OPTION

The PPU path manager may be run in one of two modes as follows:

- Dedicated mode - The path manager will always occupy a dedicated PPU whenever the operating system has at least one free pool PPU.
- Nondedicated mode - The path manager will drop the PPU and go into PP recall for the installation defined time at the end of each pass of the path manager activities.

This option is controlled by the current state of sense switch 2 (ONSW2); that is, with switch 2 ON the "dedicated mode" is selected.

RHFAM DETAIL STATUS LOGGING OPTION

The installation may select whether the subsystem will log NAD detail status errors in the system dayfile by the setting of sense switch 3 (ONSW3). When switch 3 is OFF (OFFSW3) NAD detail status errors will be logged in the dayfile.

NAD AUTODUMP

The DMPNAD control statement provides the capability to dynamically autodump the NAD memory. The calling job must be system origin or the user must be validated for system origin privileges, and the system must be in engineering mode (refer to the NOS Operator's Guide).

DMPNAD reads the NAD memory via the PP program NNC and formats the data into an output file. DMPNAD will issue appropriate messages to indicate the success or failure of the autodump attempt.

The format of the control statement when dumping a local NAD is:

DMPNAD,C=cc,L=bb,NA=aa,NL=dd.

The format of the control statement when dumping a remote NAD is:

DMPNAD,C=cc,L=bb,ND=nn,AC=aaaa,LT=ll.

- cc Channel of the local NAD.
- bb Output file name. (optional)
- aa NAD first word address. (optional)
- dd NAD last word address. (optional)
- nn Remote NAD number.
- aaaa Access code. (optional)
- ll Local trunk enable mask.

REMOTE HOST FACILITY OPERATOR UTILITY (RHFOU)

RHFOU provides the operator interface to the RHF subsystem via the DSD K-display.

This utility is invoked via the DSD command X.RHFOU. RHFAM must be active for all displays except the logical IDs (LIDT).

RHFOU allows the operator the capability to display and change parameters for the RHF application program and the network configuration.

The utility has three displays which are selected via the following commands:

K.NADT	Network Address Table
K.APPL	RHF Applications
K.LIDT	Logical IDs

The network address table defines the relationships among the EST entries and their associated LIDs. It also specifies the NAD device number (the hardware unit number on the NAD itself), the access code, the device address at the destination NAD (which applies only to the DEC mini-computer NAD interface), the remote trunk and local trunk logical mapping for the device, and the status of that logical path.

The RHF applications table describes the status of all possible RHF subsystem application programs. It also specifies the name of the job that is currently active on each application. General status information is also available.

The logical identifier table describes the general status of all mainframes as identified by their LIDs. This information includes identifying which LID is the host, all linked LIDs and whether that LID is enabled or disabled for RHF.

NETWORK ADDRESS TABLE DISPLAY

*** NETWORK ADDRESS TABLE (NAT) ***

ORD	LID	EQ	ND	AC	DD	RT	LT	ST
1	TS1	41	7	FOFO	0	10	10	E
2	TS1	40	7	FOFO	0	10	10	E
3	TS1	41	6	FOFO	0	10	10	E
4	TS1	40	6	FOFO	0	10	10	E
5	M20	40	F	FOFO	0	4	10	D

**** command entered ****

DESCRIPTION OF THE NADT DISPLAY

The Network Access Table defines the hardware and logical connections used to route messages through the LCN. The LT and RT (local trunk and remote trunk) parameters define the logical mapping of the accesses to the trunk itself from the NAD. These are hardware connections that are reflected in this table, the table being set up at installation time.

The access code is a code used by the software which is placed in the header of all NAD message traffic. The code also is checked by the NAD hardware. Its purpose is to control access to the LCN.

NETWORK ADDRESS TABLE COMMAND

The following command is available:

NAT,ord,EQ=ee,ND=nn,AC=aaaa,DD=d,RT=rr,LT=ll,ST=s.

This command must be preceded by "K."

ord	Ordinal in the Network Address Table (required parameter)
ee	EST ordinal of local NAD.
nn	Destination NAD address.
aaaa	Access code.
d	Device address at the Destination NAD.
rr & ll	Trunk control unit enables.
s	Enabled or Disabled state indication (E or D).

All parameters except ord are optional. If a particular parameter is omitted, it will not change the current setting in the Network Address Table.

Command "+." or "+" will display the next page of the table.

APPLICATION TABLE DISPLAY

*** RHFAM APPLICATION TABLE ***

ORD	NAME	JOB	CON	MC	D	I	S	A
1	QTF		0	0	N	N	N	N
2	QTFS		0	0	N	Y	N	N
3	QTFS		0	0	N	Y	N	N
4	QTFS		0	0	N	Y	N	N
5	QTFS		0	0	N	Y	N	N
6	QTFS		0	0	N	Y	N	N
7	QTFS		0	0	N	Y	N	N
10	QTFS		0	0	N	Y	N	N
11	QTFS		0	0	N	Y	N	N
12	PTFU		0	0	N	N	N	N
13	PTFU		0	0	N	N	N	N

ORD - Ordinal
NAME - Appl name
JOB - Job Seq No.
I - Auto-startable
S - Auto-started
CON - Conn active
MC - Max conn
D - Disabled
A - NETON active

**** command entered ****

DESCRIPTION OF THE APPL DISPLAY

The information for the APPL display resides in RHFAM's field length. It is a list of legal applications that RHF will allow to "NETON".

APPLICATION TABLE COMMANDS

The following commands are available:

ENABLE,xx.
DISABLE,xx.

These commands must be preceded by "K.".

These commands will enable/disable the application with ordinal xx in the table.

The command "+." or "+" will display the next page of the table.

LOGICAL ID TABLE

*** LOGICAL ID (LID) TABLE 12020

LID	AT	LID	AT	LID	AT	LID	AT	LID	AT	LID	AT
MO9	H--	Z09	H--	TS1	-L-	TS2	-L-	AAA	--D	M20	-LD

**** command entered ****

DESCRIPTION OF LID DISPLAY

Any machine in an RHF network can possess two attributes. They are:

1. Host mainframe/linked mainframe
2. Enabled/disabled

In a two machine network (machines "A" and "B"), the LIDT display on "A" will define "A" as the host and "B" as the linked mainframe, and conversely from the LIDT display on "B".

It should be noted that the first two "LIDs" defined on any machine are Mid and Zid where "id" is the machine identifier as defined in the IPRDECK. The ZID is used by the operating system for two purposes: 1) Recovery of queue files assigned to the QTF processor. 2) Identification of the servicer application as a FIP= processor at job completion time.

LOGICAL ID TABLE COMMAND

The following command is available:

SA,lid,x. (Set Attribute command)

lid Three-alphanumeric character logical identifier.

x The new attribute as follows:

- L - LINKED LOGICAL ID.
- H - HOST LOGICAL ID.
- D - DISABLE LOGICAL ID.
- E - ENABLE LOGICAL ID.

(If x is null, all attributes will be cleared. That is, the lid will be an enabled, non-host, non-linked logical identifier.)

This command must be preceded by "K."

MAINTENANCE LOGGING TRANSFER FACILITY (MLTF)

One CYBER 170 or CYBER 205 host must be identified as the "maintenance host" on each LCN. If the CYBER 170 is selected, it must have a copy of MLTF running at all times. This can be initialized by the operator as follows:

X.MLTF.

MLTF will periodically poll all NAD's that are defined in the COMSRHF NADT and enter the error log of each NAD into the Binary Maintenance Log (BML).

ACCOUNTING SUMMARY

This section provides a summary of the accounting mechanism provided by RHF in order to allow the installations to bill users in an RHF/LCN environment. The following summarizes the capabilities provided:

- Accounting messages will be kept as close as possible to the executing job and mainframe in order to reduce the need for cross processing of accounting files for multiple machines.
- Where cross processing of accounting files is required, correlation accounting entries will be made which will allow billing back to the executing job.
- New accounting entries will be made for queue and permanent file link transfers to allow billing for link traffic.

ACCOUNTING MESSAGES

Two new accounting messages are defined to allow for accounting of link usage and to coordinate accounting between linked MFs. The following accounting messages are defined for link usage:

. Correlation Message

This accounting entry can be used to coordinate account dayfiles of linked MFs.

JOBNAMEO. ACLK, JOBNAMM, PID, LID, ERR.

. File Size Message

This accounting entry indicates number of PRUs sent or received over link.

JOBNAMEO. UCLS, TY, xxxxxx.xxxKPRS.

JOBNAME	Name of job on MF for which the accounting entry is being made
0	One character identifier indicating job origin type
JOBNAMM	Name of job on linked MF (PID), this field may be blank *
PID	PID of linked MF, this field may be blank *
LID	Logical ID of MF (ST parameter), this field may be blank *
TY	Type of file
	IN Input
	PR Print
	PU Punch
	PF Permanent File
xxxxxxx.xxx	Number Of kilo PRUs
ERR	If ERR is present, the output file was discarded because the user limits were reached

* The field will be blank when the job originates at the same mainframe for execution (local batch).

INPUT FILE STAGING ACCOUNTING

Accounting messages for input files are made as follows:

1. At the sending (origin) MF, QTF issues to the account file the correlation and file size accounting messages after the job has been transferred to the linked (execution) MF.
2. At the receiving (execution) MF, job initiation issues to its account dayfile the correlation and file size accounting messages.

OUTPUT FILE STAGING ACCOUNTING

Accounting messages for output files are made as follows:

1. At the sending (execution) MF, QTF issues to the account file the correlation and file size accounting messages after the file has been sent back to the receiving (origin) MF or possibly some other MF.
2. At the receiving (origin) MF, QTFS issues to its account dayfile the correlation and file size accounting messages after the file has been received and put into the output queue.

NOTE - QTF will not transfer actual Lines Printed and Cards Punched accounting entries back to the executing MF. If an installation wants to bill for actual Lines Printed and Cards Punched, it can use the correlation entries made by QTF to combine or cross process dayfiles on both the executing and printing/punching MFs.

The following is an example of the messages generated as a result of ROUTE(...ST=TS1)

ORIGIN MAINFRAME (PID=M20)
07.40.58. AQ2YABEB. ACLK, AQ2YABQ, M09, TS1.
07.40.58. AQ2YABEB. UCLS, PR, 0.003KPRS.

REMOTE MAINFRAME (PID=M09, LID=TS1)
07.41.02. AQ2YABQS. ACLK, AQ2YABE, M20, TS1.
07.41.02. AQ2YABQS. UCLS, PR, 0.003KPRS.

PERMANENT FILE STAGING ACCOUNTING

The following accounting is provided:

- Because linked PF operations on the requesting MF will be performed by a utility which executes within the user job, the resources used to initiate and terminate file transfer will be charged directly to the user.
- Because linked PF operations on the linked MF will be performed by a job which executes under a FM/UN specified on the USER statement, file transfers will also be charged directly to the user (although the FM/UN, and so on, may be different from that of the requesting job). All other mechanisms for controlling and accounting for PF space will apply to the LINKED job in the same way that they would apply to any normal user job.
- After a linked file transfer is completed PTF and PTFS will make a file size accounting entry at their own system account file.

The following is an example of the messages generated as a result of MFLINK(...,ST=TS1,...).

```
ORIGIN MAINFRAME (PID=M20)
07.36.19. AJJYAAYB. ACLK, PTFSABF, M09, TS1.
07.37.00. AJJYAAYB. UCLS, PF, 1.807KPRS.
```

```
REMOTE MAINFRAME (PID=M09, LID=TS1)
07.36.19. PTFSABFB. ACLK, AJJYAAY, M20, TS1.
07.37.00. PTFSABFB. UCLS, PF, 1.807KPRS.
```

VALIDATION SUMMARY

This section contains a summary of the validation options provided by RHF such that the installation may control the network access and usage. Users must be validated via MODVAL for the following types of access:

- *CUST* - User can use ST on job statement or ROUTE control statement.
- *CQLK* - User can use network for file transfers.
- *CPLK* - User can use network for permanent file operations.

These options are kept in the users access control word (AAWC) and each user may examine their current setting via the LIMITS control statement.

NETWORK FAILURE PROCESSING

The normal procedure for terminating the network is to enter the DSD command n.IDLE. However, there may be times when a network program fails or the entire network fails. If the failing program is QTF, QTFS, or PTFS, a dump is automatically generated.

When the Network Access Device (NAD) fails, the operator may initiate dumping of the NAD. The following DSD command dumps the failing NAD:

```
X.DMPNAD(C=xx)
```

xx Channel number of NAD

After all necessary dumps have been taken, the operator can reload the NAD with the following command. (The system must be in Engineering Mode.)

```
X.LOADBC(C=xx)
```

xx Channel number of the NAD

If the system is configured with more than one NAD, it is possible to dump and restart just one NAD. This procedure only affects those operations using the failing NAD. All other operations in nonfailing NADs proceed normally.

1. Turn OFF the failing NAD in the EST.

```
OFFxx.
```

xx EST ordinal of NAD.

2. Dump the NAD memory (optional).

```
X.DMPNAD(C=xx)
```

3. Reload NAD memory. (The system must be in Engineering Mode.)

```
X.LOADBC(C=xx)
```

4. Turn ON the NAD in the EST.

```
ONxx.
```

xx EST ordinal of NAD.

FACILITY INTERFACE PROGRAM (FIP)

FIP is a CPU relocatable set of modules which supports a subset of the entry points which are defined by the Network Host Products Application Interface Program (AIP). This interface requires that the application program have both SSJ= and FIP= entry points and the program must be installed on the system library.

MODULE DESCRIPTIONS

The FIP modules are a part of the system library and can be accessed from the RHFIO library. All of the FIP modules have a doubly defined entry point. Either entry point may be referenced with the same results. The following is a list of equivalent entry points:

NETON	RHFON
NETOFF	RHFOFF
NETWAIT	RHFWAIT
NETGET	RHFGET
NETGETL	RHFGETL
NETPUT	RHFPUT
NETXFR	RHFREFR
NETDBG	RHFDBG

NETON

Connect the application to the NETWORK.

The FORTRAN calling sequence for NETON follows.

CALL NETON (aname,nsup,status,minacn,maxacn)

aname An input parameter specifying in display code the name of the application program, as it is identified for login. This can be one to seven 6-bit alphabetic and numeric characters, but the first must be alphabetic. This parameter must be left-justified, with blank fill. It is advisable to avoid names beginning with the letters RHF or NET to make loader map interpretation easier. The following application program names are reserved for internal networks use:

ALL	IAF	MCS	NS	QTF
BYE	ITF	NAM	NUL	QTFS
CS	LOGIN	NIP	NVF	RBF
DOP	LOGOUT	NOP	PTFU	TAF
HELLO	LOP	NOPLP	PTFS	TVF

nsup A return parameter; nsup is the symbolic address of the supervisory status word for communication from FIP to the application program. This word has the format shown below. The upper 3 bits of this word are not used. The next 2 bits are set when indicated in the figure to report the status of the data message and supervisory message queuing performed by FIP. This word need not contain zeros at the time of the NETON call and should not be changed at anytime by the application program.

nsup

c	a	n	i	s	Unused
---	---	---	---	---	--------

- c Reserved for CDC use.
- a Reserved for CDC use.
- n Reserved for CDC use.
- i Input in queue bit. This bit is set to 1 if FIP has either data messages or synchronous supervisory messages queued for the application. The bit is valid only after the application issues a NETWAIT call. After any other FIP procedure call, this bit is not necessarily correctly set. The setting of this bit is described in more detail in the description of NETWAIT later in this section.
- s Supervisory message in queue bit. This bit is set to one on return from FIP routine NETGET, NETGETL, NETPUT and NETWAIT if asynchronous supervisory messages are queued on application connection number 0 for this program. A value of one is advisory only; no program action is required. This bit is set to zero on return from NETGET (for an acn value of zero) or a return from NETGETL (for an ALN value of zero) when no asynchronous supervisory messages remain queued for the program.
- status A return parameter; status is the symbolic address of the NETON call status word. On return from the call the content of this word indicates the network software's disposition of the application program's NETON attempt. The values of status can be:
- 0 NETON was successful and network access is established.
- 1 NETON was unsuccessful because RHFAM was not at a control point or did not have enough resources to service this application program (too many application programs running at the same time).

- 2 NETON was rejected because too many application programs are presently accessing the network with the same name as specified for the aname parameter.
- 3 NETON was rejected because the application program has a status of disabled. The program must be rerun after the local operator has enabled it.
- 4 NETON was rejected because of an illegal application name. The calling application will be aborted by FIP.

minacn An input parameter specifying the smallest application connection number the application program can process; $0 \leq \text{minacn} \leq \text{maxacn} \leq 4095$. The network software assigns acn values to connections, beginning with the number specified for minacn.

maxacn An input parameter specifying the largest application connection number the application program can process; $0 \leq \text{minacn} \leq \text{maxacn} \leq 4095$. The network software does not attempt to complete any more connections to the program after all connections from minacn through maxacn (inclusive) are in use.

NOTE: The range between minacn and maxacn cannot exceed 8. The maximum number of application-to-application connections is 8 for each application.

NETOFF

Terminate connection to the NETWORK.

The FORTRAN calling sequence for NETOFF follows.

```
CALL NETOFF
```

There are no parameters.

NETWAIT

Wait until input is available or for a specified time.

The FORTRAN calling sequence for NETWAIT follows.

CALL NETWAIT(time,flag)

time An input parameter $1 \leq \text{time} \leq 4095$, specifying the number of seconds for which the application program should be suspended. If a value of zero is declared, a default value of one is used; if a value greater than 4095 is declared, a default value of 4095 is used.

flag An input parameter specifying the conditions under which processing should be resumed. This parameter can have the values:

0 Return from NETWAIT call (resume processing) when input is available from any connection or when the period declared by the time parameter has elapsed. When a flag value of zero is declared and input is available immediately, the value declared for the time parameter is ignored.

1 Return from NETWAIT call (resume processing) when the period declared by the time parameter has elapsed, regardless of whether input is available from any connection.

NETGET

Read a message from the specified connection.

The FORTRAN calling sequence for NETGET follows.

```
CALL NETGET(acn,ha,ta,tlmax)
```

acn An input parameter specifying the application connection number of the logical connection from which a message block is requested. This parameter can have the values:

0 Transfer one asynchronous supervisory message.

$\text{minacn} \leq \text{acn} \leq \text{maxacn}$ Transfer one data block or synchronous supervisory message from the logical connection with the indicated acn.

ha A return parameter specifying the symbolic address of the application program's header area. The header area always contains an updated application block header after return from the call.

ta A return parameter specifying the symbolic address of the first word of the buffer array constituting the text area for the application program. On return from the call, the text area contains the requested block if a block was available and the text area was large enough. The text area identified by ta should be at least tlmax words long.

tlmax An input parameter specifying the maximum length in central memory words of a block the application program can accept. The value declared for tlmax should be less than or equal to the length of the text area identified in the same call; if tlmax is greater than the actual length of the text area, the block transfer resulting from the NETGET call might overwrite a portion of the program. The maximum value needed for tlmax is a function of the block size used by the connection for input to the program and of the application character type the program has specified for input from the connection. The following ranges are valid:

act=1 $1 \leq t_{lmax} \leq 410$ for 60-bit (one per word) transparent characters (Note: the maximum length supported via LCN is 2043 8-bit bytes on the network)

act=2 $1 \leq t_{lmax} \leq 273$ for 8-bit (7.5 per word) ASCII characters

act=3 $1 \leq t_{lmax} \leq 410$ for 8-bit (5 per word) ASCII characters

act=4 $1 \leq t_{lmax} \leq 205$ for 6-bit (10 per word) display code characters

A t_{lmax} value of 0 can be legally declared but always results in an input-block-undeliverable condition; that is, an application block header is returned with a set ibu field, even when an empty block of application block type 2 is queued (a block with a tlc value of 0).

NETGETL

Read a message from the specified list of connections.

The FORTRAN calling sequence for NETGETL follows.

CALL NETGETL(aln,ha,ta,tlmax)

aln An input parameter specifying the number of the connection list to be scanned for a queued block. This parameter can have the values:

 0 Obtain all supervisory messages queued on application connection number 0 first and then any data or synchronous supervisory message blocks queued on another connection.

$1 \leq aln \leq 63$ Obtain one data block or synchronous supervisory message from one connection on the indicated list.

ha A return parameter, as input to the call, is the symbolic address of the application program's header area. The header area always contains an updated application block header after return from the call.

ta A return parameter, as input to the call, is the symbolic address of the first word of the buffer array constituting the text area for the application program. On return from the call, the text area contains the requested block if a block was available and the text area was large enough. The text area identified by ta should be at least t1max words long.

t1max An input parameter specifying the maximum length in central memory words of a block the application program can accept. The value declared for t1max should be less than or equal to the length of the text area identified in the same call; if t1max is greater than the length of the text area, the block transfer resulting from the NETGETL call might overwrite a portion of the program. The maximum value needed for t1max is a function of the block size used by the connection for input to the program and of the application character type the program has specified for input from the connection. The following ranges are valid:

act=1 $1 \leq t1max \leq 410$ for 60-bit (one per word) transparent characters (Note: the maximum length supported via LCN is 2043 8-bit bytes on the network)

act=2 $1 \leq t1max \leq 273$ for 8-bit (7.5 per word) ASCII characters

act=3 $1 \leq t1max \leq 410$ for 8-bit (5 per word) ASCII characters

act=4 $1 \leq t1max \leq 205$ for 6-bit (10 per word) display code characters

A t1max value of 0 can be legally declared but always results in an input-block-undeliverable condition; that is, an application block header is returned with a set ibu field, even when an empty block of application block type 2 is queued (a block with a tlc value of 0).

NETPUT

Write a message to a specified connection.

The FORTRAN calling sequence for NETPUT follows.

```
CALL NETPUT(ha,ta)
```

- ha An input parameter specifying the symbolic address of the application program's block header area. The block header area must contain a valid application block header word.
- ta An input parameter specifying the symbolic address of the application program's text area. The text area must contain a valid data message or supervisory message block, correctly described by the contents of the block header area.

NETXFR

Start a file transfer on a specified connection.

The FORTRAN calling sequence for NETXFR follows.

```
CALL NETXFR (acn,fname,op,status,wait,dd,tmout)
```

- acn Application connection number $acn_{min} < acn < acn_{max}$
- fname An input parameter specifying in display code the name of the file to be transferred.
- op An input parameter specifying whether the operation is a read or a write across the network.
- 0 The file is to be read from the network and written to disk.
- 1 The file is to be read from disk and written to the network.

status A return parameter; status is the symbolic address of the NETXFR call status word.

0 The file transfer is in process.

1 The transfer is complete (no error).

2 The file transfer was completed with a network error.

wait An input parameter specifying the condition under which processing should resume.

0 Return from NETXFR call (resume processing) when the file transfer is completed.

1 Return from NETXFR call (resume processing) immediately. The application program must monitor the status word in order to determine the progress of the file transfer.

dd File transfer data declaration which is interpreted by the initiating host, and transmitted, via protocol, to the linked host.

0 - US The file contains undefined data in a logical structure.

1 - UU The file contains undefined data in an undefined format.

2 - C6 The file contains character data, in a 64-character (or less) set.

3 - C8 The file contains character data, in a character set of more than 64 characters.

tmout An input parameter specifying the time to wait for a message response before timing out.

NTPXFR

For a NETXFR transfer where the immediate return option is selected (WAIT=1), the application will receive the protocol messages (6 and 7) and the supervisory messages required by NETXFR. To transfer these messages to NETXFR for their appropriate actions it is necessary to have a reentry point to NETXFR. This entry is called NTPXFR.

The FORTRAN calling sequence for NTPXFR follows.

```
CALL NTPXFR(ha,ta)
```

ha	An input parameter specifying the symbolic address of the messages header area. The block header area must contain a valid application block header word.
ta	An input parameter specifying the symbolic address of the application's message text area. The text area must contain a valid data message or supervisory message block. It must be correctly described by the contents of the block header area and must be associated with an ACN with an active NETXFR transfer.

NETDBG

Supervisory and/or data message flow through the program can be traced by optional FIP code; this code makes entries in the dayfile for such messages. The optional FIP code that makes the dayfile entries gives an application program a means of recording all exchanges between the program and the network. The FIP utility routine NETDBG gives the program a method of selecting exchanges which should be recorded.

Whether or not the log file is created depends on how system library (RHFIO) used to satisfy the application program's externals was built. The code which makes the dayfile entries is referred to as the logging code. When a version of RHFIO with the logging code built into it is used, all code needed to make the dayfile entries is loaded; the options for logging both supervisory messages and data messages are automatically turned off initially. Because this logging code causes additional processing overhead and central memory requirements for the application program's control point, it should be removed after the program is completely debugged. The code can be removed from the job without altering the application program's structure by loading the FIP code from a version of RHFIO

without the logging code built into it. When such a version is used, the only part of the logging code loaded is a do-nothing version of NETDBG.

To build a version of FIP with the logging code, use the *DEFINE FIPDB Modify directive (refer to RHF Application Installation Options in section 4).

Calls to NETDBG can occur in programs using either version of RHFIO. For example, when a NETDBG call turns either or both supervisory and data message logging on and a status is returned indicating logging is not possible, no error occurs and the option selection is ignored.

The FORTRAN calling sequence for NETDBG follows.

CALL NETDBG(dbugsup,dbugdat,avail)

dbugsup An input parameter that turns the logging of supervisory messages on or off. This parameter can have the values:

 =0 turn supervisory message logging on.

 ≠0 turn supervisory message logging off.

When supervisory message logging is turned on, all supervisory messages between the application program and RHFAM are logged. Logging occurs whenever a call to NETGET, NETGETL, or NETPUT causes a message transfer. This logging continues until a call with a nonzero debugsup parameter is issued.

dbugdat An input parameter that turns the logging of data messages on or off. This parameter can have the values:

 =0 turn data message logging on.

 ≠0 turn data message logging off.

When data message logging is turned on, the first central memory word of all data messages exchanged on any connection between the application program and RHFAM are logged. Logging occurs whenever a call to NETGET, NETGETL, or NETPUT causes a message transfer. This logging continues until a call with a nonzero dbugdat parameter is issued.

avail

A return parameter that indicates whether or not the logging code portion of was built into RHFIO. On return from the call, this parameter can have the values:

=0 logging code was built and logging is possible.

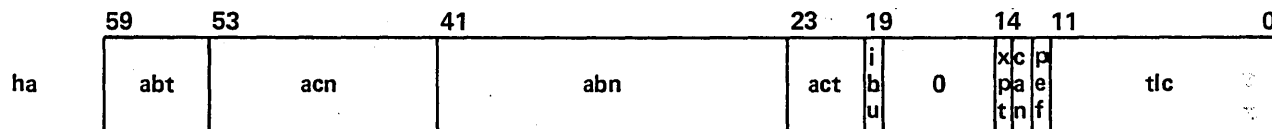
=1 logging code was not built and logging is not possible

When a value of 1 is returned, specification of 0 for either dbugsup or dbugdat has had no effect, but did not cause an error.

DATA BLOCK HEADER FORMATS

The format of the block header word associated with a data block depends on whether the block is being sent or received by the application program.

Application Block Header Format for Input Data Blocks



ha Symbolic header area address specified as the location receiving the application block header in a call to NETGET or NETGETL.

abt Application block type of the associated data block. This field can have the values:

=0 Indicates a null block. (No block is queued from the logical connection polled.)

=1 Indicates the associated block is one of several blocks comprising a single message but is not the last such block.

=2 Indicates the associated block is either the last or the only one comprising the message.

Values of 3 through 63 are not valid for data blocks on input. This field can be accessed with the reserved symbol ABHABT.

acn Application connection number of the logical connection from which the associated data block was received. This field can have the values $1 \leq \text{minacn} \leq \text{acn} \leq \text{maxacn} \leq 4095$, where the values minacn and maxacn are parameters in the NETON statement. This field can be accessed with the reserved symbol ABHADR.

abn Application block number assigned to the associated data block. If the application block header is associated with a data block from another application program, this field contains the 18-bit integer used by that program to identify the block when the block was sent. This field can be accessed with the reserved symbol ABHABN.

act Application character type used to encode the accompanying data block. This field can have the values:

- =1 60-bit transparent characters, packed one per central memory word.
- =2 8-bit character, packed 7.5 per central memory word.
- =3 8-bit character, right justified, in 12-bit bytes with zero fill, packed 5 per central memory word.
- =4 6-bit display code characters, packed 10 per central memory word.

Values of 0 and 5 through 15 are reserved for expansion and presently are not valid. The value contained in the act field is the value assigned to the connection by the application program for input in the connection-accepted supervisory message. This field can be accessed with the reserved symbol ABHACT.

ibu Input-block-undeliverable bit. When this bit is set (has a value of 1), the data block associated with this block header has not been delivered to the application program. This field can be accessed with the reserved symbol ABHIBU. A set ibu bit normally indicates that the block is larger than the maximum text length (tlmax parameter) declared by the application program in its NETGET call and remains queued by FIP until a call is issued that specifies an adequate text length. The block header for such a block contains the actual length of the queued block in its tlc field, given in character units specified by the act field.

xpt Reserved for future use. This field can be accessed with the reserved symbol ABHXPT.

can Reserved for future use. This field can be accessed with the reserved symbol ABHCAN.

pef Reserved for future use. This field can be accessed with the reserved symbol ABHBIT.

tlc Text length of the associated data block, in character units specified by the act parameter. The equivalent length in central memory words can be computed as follows:

act=1 tlc is the number of central memory words the block can occupy.

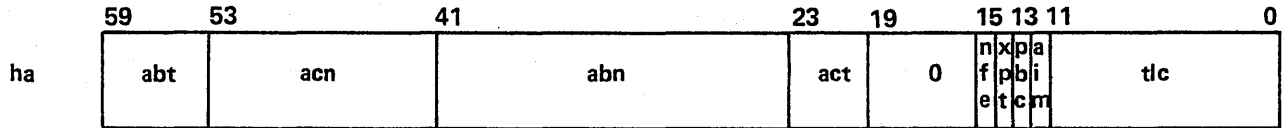
act=2 The number of central memory words the block can occupy is tlc divided by 7.5, rounded upward to an integer.

act=3 The number of central memory words the block can occupy is tlc divided by 5, rounded upward to an integer.

act=4 The number of central memory words the block can occupy is tlc divided by 10, rounded upward to an integer.

This field can be accessed with the reserved symbol ABHTLC.

Application Block Header Format for Output Data Blocks



ha Symbolic header area address, specified as the application block header's location in a call to NETPUT.

abt Application block type of the accompanying data block.
This field can have the values;

- =1 Indicates the accompanying block is one of several blocks comprising a single message but is not the last such block.
- =2 Indicates the accompanying block is either the last or only one comprising a message.

Values of 0 and 3 through 63 are not valid for data blocks on output. This field can be accessed with the reserved symbol ABHABT.

acn Application connection number of the logical connection to which the accompanying data block should be sent. This field can have the values $1 \leq \text{minacn} \leq \text{acn} \leq \text{maxacn} \leq 4095$, where the values minacn and maxacn are parameters in the NETON statement. This field can be accessed with the reserved symbol ABHADR.

abn Application block number assigned to the data block being sent. This field is an 18-bit integer that identifies the block when the network software's processing of the block returns certain supervisory messages. The definition of the block number is left to the programmer for maximum flexibility; it can be:

A sequencing number.

The block's central memory address.

The block's mass storage address (physical record unit).

An index value for a block control array or table.

An external label.

This field can be accessed with the reserved symbol ABHABN.

act Application character type used to encode the accompanying data block. This field can have the values:

- =1 60-bit transparent characters, packed one per central memory word.
- =2 8-bit characters, packed 7.5 per central memory word.
- =3 8-bit characters, right-justified in 12-bit bytes, packed 5 per central memory word.
- =4 6-bit display code characters, packed 10 per central memory word.

Values of 0 and 5 through 15 are reserved and presently are not valid. This field can be accessed with the reserved symbol ABHACT.

nfe Reserved for future use.

This field can be accessed with the reserved symbol ABHNFE.

xpt Reserved for future use.

This field can be accessed with the reserved symbol ABHXPT.

pbk Reserved for future use.

This field can be accessed with the reserved symbol ABHCAN.

aim Reserved for future use.

This field can be accessed with the reserved symbol ABHBIT.

tlc Text length of the associated data block, in character units specified by the act parameter. The value to use in the tlc field can be computed as follows:

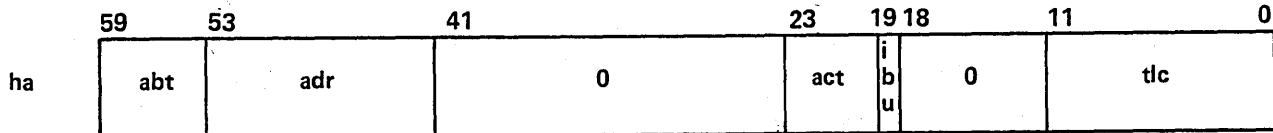
- act=1 tlc is the number of central memory words occupied by the block.
- act=2 tlc is the number of complete central memory words occupied by the block times 7.5, plus the number of complete character bytes used in the remaining central memory word, rounded upward to an integer.
- act=3 tlc is the number of complete central memory words occupied by the block times 5, plus the number of 12-bit character bytes used in the remaining central memory word.
- act=4 tlc is the number of complete central memory words occupied by the block times 10.

This field can be accessed with the reserved symbol ABHTLC.

SUPERVISORY BLOCK HEADER FORMATS

The format of the block header word associated with a supervisory message depends on whether the message is asynchronous or synchronous and on whether it is being sent or received.

Application Block Header Format for Input Supervisory Messages



ha Symbolic header area address specified as the location receiving the application block header in a call to NETGET or NETGETL.

abt Application block type of the associated message block. This field can have the values:

=0 Indicates a null block. (No supervisory message is queued from the logical connection polled.)

=3 Indicates the accompanying block is a supervisory message block.

Values of 1, 2, and 4 through 63 are not valid for supervisory messages on input. This field can be accessed with the reserved symbol ABHABT.

adr Application connection number of the logical connection from which the message block comes. This field can have the values:

=0 For asynchronous supervisory messages from the host portion of the network software.

=acn Reserved for future use.

This field can be accessed with the reserved symbol ABHADR.

act Application character type used to encode the accompanying message block. The value appearing in this field depends on the type of supervisory message involved; this field can have the values:

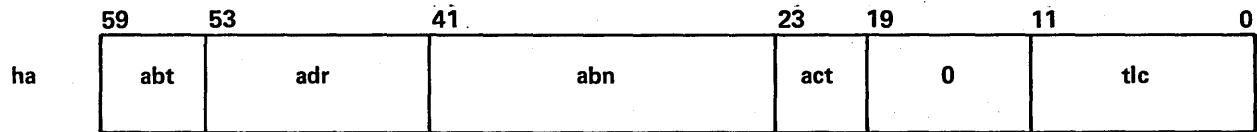
- =1 An asynchronous supervisory message, packed in 60-bit transparent characters (one character per central memory word).
- =2 A synchronous supervisory message packed in 8-bit ASCII characters (7.5 characters per central memory word).

Because the fields within the supervisory messages are defined as groups of bits within central memory words (rather than as characters in a character string), the act parameter of a supervisory message does not indicate that character mapping occurred. This field can be accessed with the reserved symbol ABHACT.

ibu Input-block-undeliverable bit. When this bit is set (has a value of 1), the message block associated with this block header has not been delivered to the application program. The block is larger than the maximum text length (tlmax parameter) declared by the application program in its NETGET or NETGETL call and remains queued by FIP until a call is issued that specifies an adequate text length. A block header containing a set ibu bit contains the actual length of the queued block in its tlc field, given in character units specified by the act field. This field can be accessed with the reserved symbol ABHIBU.

tlc Text length of the associated message block, in character units specified by the act parameter. If act is 1, tlc is the number of central memory words occupied by the message block; if ACT is 2, tlc is the number of 8-bit bytes containing meaningful message fields. This field can be accessed with the reserved symbol ABHTLC.

Application Block Header Format for Output Supervisory Messages



ha Symbolic header area address specified as the application block header's location in a call to NETPUT.

abn Application block type; abn is 3 for all supervisory messages. This field can be accessed with the reserved symbol ABHABT.

adr Application connection number of the logical connection to which the message block should be sent. This field can have the values:

=0 For asynchronous supervisory messages addressed to the host portion of the network software.

=acn Reserved for future use.

This field can be accessed with the reserved symbol ABHADR.

abn Application block number assigned to the message block being sent. This field is an 18-bit integer that identifies the block when the network software's processing of the block returns a block-delivered supervisory message (asynchronous supervisory messages are not acknowledged by this method). The definition of the block number is left to the programmer for maximum flexibility; it can be:

A sequencing number.

The block's central memory address.

The block's mass storage address (physical record unit).

An index value for a block control array or table.

An external label.

This field can be accessed with the reserved symbol ABHABN.

act Application character type used to encode the accompanying message block. The value declared for this field depends on the type of supervisory message involved; this field can have the values:

=1 An asynchronous supervisory message, packed in 60-bit transparent characters (one character per central memory word).

=2 A synchronous supervisory message, packed in 8-bit ASCII characters (7.5 characters per central memory word).

This field can be accessed with the reserved symbol ABHACT.

tlc Text length of the accompanying message block in character units specified by the act parameter. If act is 1, tlc is the number of central memory words occupied by the message block; if act is 2, tlc is the number of 8-bit bytes containing meaningful message fields. This field can be accessed with the reserved symbol ABHTLC.

This glossary defines terms unique to the description of the software presented in this manual. It also contains terms whose interpretation within this manual is intended to be more constrained or different from that commonly made. Some terms used in other manuals for the network software are included for the reader's convenience when reconciling terminology.

Acknowledgement, Block -

A message returned to the sender confirming the delivery of one block; referred to as BACK in CCP documentation.

Application Block Header (ABH) -

A single 60-bit word description accompanying every block passing between an application program and RHFAM.

Application Block Number (ABN) -

A field in the application block header. An application-assigned number used to identify a particular data message block.

Application Block Type (ABT) -

A field in the application block header defining the accompanying block as either data or supervisory, null or not null, and indicating if this is the last block of a message.

Application Character Type (ACT) -

A field in the application block header defining the byte size of text characters.

Application Connection Number (ACN) -

A number assigned by the Communications Supervisor program to identify a particular logical connection within an application program.

Application List Number (ALN) -

An application-program-assigned number used to identify a particular group of logical connections belonging to the application program.

Application Name (ANAME) -

Up to seven 6-bit letters or digits (the first must be a letter) used to identify an application program. It is used by another application program or by a terminal operator when connection to the application is requested.

Application Program -

A program resident in a host computer that provides an information storage, retrieval, and/or processing service via the data communication network and the RHF Access Method. In the context of network software, an application program is not an interactive job, but rather a file transfer servicing facility. A file transfer servicing facility provides users with a specific processing capability such as routing a file to a queue on a linked mainframe or retrieving a file from a remote mainframe.

Block -

In the context of network communications, a portion or all of a message. A message is divided into blocks to facilitate buffering, transmission, error detection and correction of variable length data streams.

Block Acknowledgment -

See Acknowledgment, Block.

Block Header -

See Application Block Header.

Block Type -

See Application Block Type.

Break -

A method employed by a terminal operator to interrupt output or input in progress. Also, an element of the block protocol that indicates an interruption of the data stream.

Byte -

A group of bits. Unless prefixed (for example, a 6-bit byte), the term implies 8-bit groups. When used for encoding character data, a byte represents a single character.

Character -

Unless otherwise specified, references to characters in this manual are to ASCII 8-bit byte characters.

Communication Element -

Any entity that constitutes a point of input to, or output from, the data communication network. This includes terminal devices, terminals, communication lines, and application programs.

Connection -

See Logical Connection.

Connection Number -

See Application Connection Number.

C6 -

File conversion mode parameter (DD=C6) meaning 6-bit character representation, structured, for the MFLINK and MFQUEUE control statements.

C8 -

File conversion mode parameter (DD=C8) meaning 8-bit character representation, structured, for the MFLINK and MFQUEUE control statements.

Data -

Any portion of a message created by the source, exclusive of any information used to accomplish transmission of such a message.

Destination -

The terminal or application program designated to receive the message.

Direct Access File -

In the context of NOS permanent files, a direct access file is a file that is accessed and modified directly.

Downline -

The direction of output flow, from host to linked computer.

Facility Interface Program (FIP) -

A group of routines that reside in the application program's field length. These routines translate and buffer communication between the application program and the network.

Header Area (HA) -

An area, usually one 60-bit word, within the application program containing the application block header for a NETPUT call, or the area to receive the header for a NETGET or NETGETL call.

Host -

A computer that executes application programs.

Input -

Information flowing upline from a linked computer to a host computer.

Input Parameter -

A parameter in an FIP call that provides input to the FIP routine. An input parameter can be a constant, an expression, or a symbolic address for such values. Input parameters are not altered by the completion of FIP processing.

LCN -

Loosely Coupled Network; also referred to in some industry standards as Local Computer Network.

List -

A group of logical connections with the same application list number, which are linked together by RHFAM and treated as a single entity in NETGETL calls.

Local Configuration File -

A file in the host computer system, containing information on the physical and logical makeup of the communication elements in the system. The file contains a list of the application programs available for execution in the host computer, and the lines and terminals that can access it. This is a NOS direct access permanent file.

Local Operator -

The administrative operator who manages the communication elements of the network within the computer system by communicating with the Communications Supervisor in the host computer. The local operator is an administrative operator within the network and need not be the host computer's operating system operator. Contrast with Network Operator.

Logical Connection -

A logical message path established between two application programs. Until terminated, the logical connection allows messages to pass between the two.

Logical Record -

Under NOS, a data grouping that consists of one or more PRUs terminated by a short PRU or zero-length PRU.

Message -

A logical unit of information, as processed by an application program. When transmitted over a network, a message can consist of one or more blocks.

MMF -

Multimainframe.

Network -

An interconnected set of network access devices.

Network Operator -

The administrative operator who manages the hardware, linkages, and other network elements of the data communication network. The network operator can also be a local operator, but might not be the operating system operator for the host computer. Contrast with Local Operator.

Output -

Information flowing downline from host to terminal.

Peripheral Processor Unit (PPU) -

The hardware unit within the host computer that performs physical input and output through the computer's data channels.

Physical Record Unit (PRU) -

Under NOS, the amount of information transmitted by a single physical operation of a specified device. The size of a PRU depends on the device, as shown in table A-1.

TABLE A-1. PRU SIZE

<u>Device</u>	<u>Size in Number of 60-Bit Words</u>
Mass storage	64
Tape in SI format with binary data	512
Tape in I format	512

A PRU that is not full of user data is called a short PRU; a PRU that has a level terminator but no user data is called a zero-length PRU.

Protocol -

A set of standardized conventions which must be used to achieve complete communication between elements in a network. A protocol can be a set of predefined coding sequences, such as the control byte envelopes added to or removed from data exchanged with a terminal; a set of data addressing and division methods, such as the block mechanism used between an application program and the RHF Access Method; or a set of procedures used to control communication, such as the supervisory message sequences used between an application program and the RHF Access Method.

PRU Device -

Under NOS, a mass storage device or a tape in SI or I format, so called because records on these devices are written in PRUs.

PTF -

Permanent File Transfer Facility; that part of RHF which processes permanent files.

QTF -

Queue File Transfer Facility; that part of RHF which processes queue files.

Random File -

In the context of the NOS operating system, a file with the random bit set in the file environment table in which individual records are accessed by their relative PRU numbers.

Return Parameter -

A parameter in an FIP call that provides as input to the FIP routine the identification of a location to which FIP should transfer information. This location is within the application program's field length and outside of the FIP portion of that field length. A return parameter cannot be a constant or a value in itself. Return parameters are always symbolic addresses. The time at which transfer of information from FIP occurs depends on whether the program is operating in parallel mode and whether use of the parameter is global to all FIP routines or local to the call in which it is used.

Sequential -

A file organization in which records are stored in the order in which they are generated.

Short PRU -

A PRU that does not contain as much user data as the PRU can hold, and this is terminated by a system terminator with a level number. Under NOS, a short PRU defines EOR.

Source -

The terminal or host computer program that creates a message.

Source Node -

The node that interfaces directly to the source of a data message block.

Supervisory Message -

A message block not directly involved with the transmission of data, but which provides information for establishing and maintaining an environment for the communication of data, between the application program and RHFAM, and through the network to a destination or from a source.

Symbolic Address -

The abstract identification of an entity serving as a location from which or to which information can be transferred. A symbolic address can contain information, but does not constitute information. A symbolic address is an identifier represented in character form by the programmer, and is equivalent to the concept of a variable in the terminology of some programming languages. In FORTRAN or ALGOL programs, typical symbolic addresses include array names, array element names, and variable names. In COMPASS, a symbolic address is equivalent to a label in a source code location field; a relative address cannot be used as a symbolic address. In COBOL, a symbolic address is equivalent to a level 01 Data Description entry. In SYMPL, a symbolic address is equivalent to the name of an array or scalar item in a data declaration.

TCU -

Trunk Control Unit.

Text Area (TA) -

The area within the application program that receives the message block text from a NETGET or NETGETL call, or contains the message block text for a NETPUT call.

Text Length in Characters (TLC) -

A field in the application block header specifying the number of character bytes of text in the message block.

Text Length Maximum (TLMAX) -

Maximum length in central memory words of the data message text block that the application program will accept for processing.

Trunk -

The communication line connecting two network access devices.

Upline -

The direction of input flow from linked to host computer.

US -

File conversion mode parameter (DD=US) meaning undefined, structured, for the MFLINK and MFQUEUE control statements.

UU -

File conversion mode parameter (DD=UU) meaning undefined, unstructured, for the MFLINK and MFQUEUE control statements.

Zero-Length PRU -

A PRU that contains system information, but no user data. Under NOS, a zero-length PRU defines EOF.

CONSOLE MESSAGES

Messages for operation of the NOS operating system with RHF are described in this appendix. Diagnostic messages are issued by routines of the NOS operating system or RHF. Standard NOS system messages are described in the NOS Diagnostic Handbook.

Messages pertaining to the operation of RHF are in the first section of this appendix. These messages appear in one or more of the following: NOS system B display, NOS system dayfile, NOS system errlog, RHFAM control point dayfile, and job dayfile.

Lowercase letters in a message indicate a variable field. Those fields are explained in the description accompanying the message.

RHF ACCESS METHOD MESSAGES

DEFAULT NETWORK ADDRESS TABLE USED.

The NADTxx file was not accessible under the system user index (377777B) as an indirect access file (xx = mainframe id). The preassembled table is used.

NADT TABLE EXCEEDS MAXIMUM SIZE.

This NADTxx file contains more entries than space was allowed for. The installation parameter NATL (see section 7.1) must be increased to accommodate the whole table. The entries which would not fit are ignored.

NC,Ccc,Fffff,Sssss,Hhhhh,Bbbbbbb,Ppppp.
NC,Ccc, , text.

An error was detected by one of the RHFAM subsystem drivers.
The type of error is indicated by the text portion of the
message as follows:

DEVICE ERROR.	The device has returned a hardware fault status.
FUNCTION TIMEOUT.	The device did not respond to a channel function.
RESPONSE ERROR.	The controlware response was not normal.
LENGTH ERROR.	The block of data transferred was incorrect in length.
ADDRESS OUT OF RANGE.	The parameter address was out of range.
CHANNEL ACTIVE ERROR.	The channel was active at initiation of a new function code.
CONTROLWARE TIMEOUT.	The flag function was not responded to.
PRIMED TIMEOUT.	The LCN hardware did not return a ready for transfer status.
CONTROL MESSAGE REJECTED.	A control message was found on the reject queue.

cc	Channel number
ffff	Current hardware function
ssss	Controlware status
hhhh	Hardware status
bbbbbb	Block number
pppp	PP P-address of detected error

FACILITY INTERFACE PROCESSOR (FIP) MESSAGES

The following error messages can be issued to the applications dayfile:

DUPLICATE CON/REQ/N FOR PATH ACN.
ACN RANGE DEFINITION ERROR.
ACN PATH NOT ACTIVE.
ACN RANGE ERROR.
INVALID APPLICATION LIST NUMBER.
INVALID BLOCK TYPE.
INVALID CHARACTER TYPE
FIP= ENTRY POINT MISSING.
NETxxxx BEFORE NETON.
NETOFF WITHOUT PREVIOUS NETON.
NETOFF WITH ACTIVE PATHS.
DUPLICATE NETON.
NETWORK FAILURE.
APPLICATION CONNECTION LOST.
RHFAM NOT RESPONDING
SUBSYSTEM NOT ACTIVE.
SUBSYSTEM ERROR XX.

PERMANENT FILE TRANSFER FACILITY (PTF) MESSAGES

The following error messages are issued to the dayfile by PTF for the reasons stated.

<u>Dayfile Message</u>	<u>Reason Issued</u>
INVALID ACCESS VALIDATION.	The user number in use does not have the required validation set.
INVALID LID.	The logical id (LID) specified is not a linked LID.
LID DISABLED.	The specified LID is not enabled for use.*
RHFAM NOT ACTIVE.	The Remote Host Facility (RHFAM) subsystem is not active.*
NETWORK FULL.	RHFAM has rejected the PTF application due to insufficient resources.*

APPLICATION DISABLED.	PTF has been disabled for use.
INVALIDxxxxxxx.	MFLINK parameter (xxxxxxx) or
INVALIDxxxxxxx=yyyyyyy.	value not valid or missing.
INVALIDxxxxxxx NO VALUE.	
INVALID COMMAND xx.	PTF received an invalid command
	(xx)from the servicer or received
	a command out of sequence.**
INVALID PARAMETER xx FOR yy.	The parameter (xx) sent with
	command (yy) is invalid.***
APPLICATION CONNECTION REJECTED.	A connection reject was received
	from the subsystem. Check LID
	and retry.*
APPLICATION CONNECTION TIMEOUT.	A response from the servicer was
	not received in the allotted
	time.**
APPLICATION CONNECTION BROKEN.	The connection with the servicer
	was broken by the NETWORK or
	servicer.**
FILE TRANSFER ERROR.	An error was encountered during
	the file transfer.****
PROTOCOL ERROR INxx.	The servicer has sent a parameter
	(xx) with an unrecognized text.***
FC/BRK RECEIVED RC=xxx.	The servicer has sent a break with
	reason code xxx.** NOS PTFS uses
	the following codes -
	0-77B = NOS error code
	101B = Connection break was
	expected but not
	received.

NOTE: The error messages given are followed by an abort with the following exceptions:

- * The error causes a rollout and retry sequence if user real-time processing (RT) is not selected.
- ** The error condition breaks and reestablishes connection for an installation-defined number of retries if EP is not selected.
- *** The dayfile message is informative only, and no abort takes place.
- **** The error condition is retried if the servicer so indicates and the EP parameter is not selected. The maximum number of file retransmissions is an installation defined value.

PERMANENT FILE TRANSFER SERVICER (PTFS) MESSAGES

The error messages issued by PTFS are either to the local dayfile or to the remote mainframe. Those sent to the remote initiator categorized as either dayfile or operator messages. The messages are issued or sent for the indicated reason. All errors cause an abort unless otherwise noted.

LOCAL DAYFILE MESSAGES

<u>Dayfile Message</u>	<u>Reason Issued</u>
APPLICATION CONNECTION BROKEN.	The connection with the initiator was broken by the NETWORK or initiator.
APPLICATION CONNECTION TIMEOUT.	The connection is being broken because the installation-defined - time-out period was exceeded while waiting for a response from the initiator.
FILE TRANSFER ERROR.	The file being transferred by NETXFR did not complete normally.*
INVALID COMMAND xx.	PTFS received an invalid command (xx) from the initiator or received acommand out of sequence.

INVALID CONTROL CARD.	PTFS was called as a control statement by a user call.
INVALID PARAMETER xx FOR yy.	An invalid parameter (xx) was sent with command (yy).*
PROTOCOL ERROR IN xx.	Parameter with attribute (xx) has an invalid associated text.*
RHFAM STATUS NOT NETTED ON.	The Remote Host Facility has terminated or the application has been disabled.
FC/BRK RECEIVED RC=xxx.	The initiator has sent a break with reason code xxx. NOS PTF uses the following codes - 0-77B = NOS error code 100B = User validation failure 102B = Echo Text Error

* Error messages are informative only and no abort takes place.

REMOTE DAYFILE MESSAGES

The following messages are sent to the initiator as dayfile type messages. The prefix PTFS- is added to each message before sending.

<u>Remote Dayfile Message</u>	<u>Reason Issued</u>
CHARGE REQUIRED.	The processing of the most recent USER statement indicates the user must supply a CHARGE statement.
FILE ALREADY PERMANENT.	The request SAVE or SAVEPF found the file already exists.
FILE IS DIRECT ACCESS.	The requested REPLACE or APPEND found a direct access file already permanent.
INVALID ACCESS VALIDATION.	The USER card being processed found the user number to be invalid or not validated for processing.

INVALID CONTROL STATEMENT.

The supplied user text was not recognized.

INVALID xxxxxxxx.

INVALID xxxxxxxx=yyyyyyy.

INVALID xxxxxxxx NO VALUE.

The user text control statement contained an invalid parameter (xxxxxxx) or value (yyyyyy).

MISSING CHARGE/PROJECT.

The CHARGE statement did not have the required charge and project numbers.

MISSING USERNUM.

The USER statement did not contain the required user number.

USER CARD REQUIRED FIRST.

NOS, and therefore PTFS, requires a user number to access permanent files.

NOTE: Additional dayfile messages of the form PTFS-xxx...xx. are issued to the remote initiator to convey PFM error messages (xxx...xx.). These messages can be found in the NOS Reference Manual, Volume 1.

REMOTE OPERATOR MESSAGES

Messages in this category are sent to the initiator as operator type dayfile messages. The prefix PTFS- is added to each message before sending.

Remote Operator Message

Reason Issued

SECOND USER TEXT ENCOUNTERED.

Two or more user text parameters were sent. Only the first is used.

USER TEXT DIRECTIVE MISSING.

No user text was supplied.

QUEUE FILE TRANSFER FACILITY MESSAGES

B DISPLAY MESSAGES.

The following messages are put on message line 2 of the B-Display only and are only informative.

IDLE	No transfers currently active
n TRANSFERS ACTIVE.	There are currently n queue files being transferred

ACCOUNT FILE MESSAGES

JOBNAMEO. ACLK, JOBNAMM, PID, Correlation Message
LID,ERR.

JOBNAMEO. UCLS, TY, File Size Message
xxxxxxx.xxxKPRS.

where:

JOBNAM = name of job on MF for which the accounting entry is being made.
O = one character identifier indicating job origin type.
JOBNAMM = name of job on linked MF (PID).
PID = PID of linked MF.
LID = Logical ID of MF (ST parameter).
ERR = If ERR is present the output file was discarded because the user limits were reached.
TY = type of file
IN: Input
PR: Print
PU: Punch
PF: Permanent File
xxxxxxx.xxx =number of kilo PRUs.

USER FILE MESSAGES

The following messages are returned to the user input files and will be printed under the users banner.

USER xxxxxxxx NOT ALLOWED TO SEND FILE yyyyyyy ACROSS THE NETWORK, QTF	User xxxxxxxx tried to send file yyyyyyy across and his validation does not allow him to do so.
USER xxxxxxxx - BAD DD PARAMETER FILE yyyyyyy, QTF	An explicit route of file yyyyyyy contained a bad DD parameter for the file.
USER xxxxxxxx - BAD EXPLICIT TEXT WITH FILE yyyyyyy, QTF	The explicit text that accompanies file yyyyyyy is invalid.
R xxxxxxxx - FILE NOT WANTED BY ALL REMOTE HOSTS, QTF	All of the remote hosts refused to accept this file.
USER xxxxxxxx - ILLEGAL USER NO. ON REMOTE HOSTS, QTF.	The specified user number was illegal on all remote hosts.
USER xxxxxxxx - NO DESTINATION PID, SEE ANALYSTS, QTF.	The lid specified on this file has no remote host PID associated with it. See analysts for resolution.

QTF/QTFS MESSAGES

The column in the center contains either a F, FF, or I. F means fatal to the queue file transfer facility, FF, fatal to the file transfer and I for informative.

APPLICATION DISABLED, QTF	F	QTF tried to neton to RHFAM and was told that it was disabled.
ATTRIBUTES WITH COMMAND xx IGNORED, pid/lid, QTF.	I	Command xx included attributes that were ignored.
CONNECTION TIMED OUT, ac, pid/lid, QTF	FF	This connection was timed out because there were no messages within the allotted time.
CONNECTION NUMBER NOT IN MESSAGE, QTF.	F	The application connection number was not in the network message.

ERROR IN FILE TRANSFER xxxxxxx, pid/lid, EC=ec, ACN=ac, QTF.	FF	An error as designated by EC occurred in transferring file xxxxxxx.
FATAL DSP ERROR (EC=xx), QTF	F	An error (xx) returned from DSP is considered fatal.
FATAL QAC ERROR (EC=xx), QTF.	F	An error (xx) returned from QAC is considered fatal.
FC/BREAK RECEIVED, RC=rc,ACN= ac, QTF.	FF	A network message "FC/Break" was received for the connection.
FC/BREAK SENT, RC=rc, ACN=ac, QTF.	FF	A network message "FC/Break" was sent for the connection.
FILE xxxxxxxx ROUTED TO LOCAL OUTPUT FROM DESTINATION ID=yyy QTF.	FF	The file xxxxxxxx was routed to the local I/O queues by QTF for one of the specified reasons (refer to USER FILE MESSAGES for reasons and error messages).
ILLEGAL RESPONSE, QTF.	F	NETON returned an unrecognized response.
INVALID LID FOUND IN QTFTID FILE, QTF, xxxxx.	I	An invalid LID xxxxx was found in the PID/LID file.
INVALID PID FOUND IN QTFTID FILE, QTF, xxxxx.	I	An invalid PID xxxxx was found in the PID/LID file.
INVLAID SEQUENCE RECEIVED, pid/lid, QTF. xxxxxxxxxxxx yyyyyyyyyyy	FF	An invalid sequence of level 7 messages was received.
LFM ERROR xx IN ASSIGNING FILE TC QUEUE DEVICE, QTF.	FF	The Queue File servicer encountered the error when trying to assign the file to a Queue device.
MESSAGE RECEIVED IS NOT RECOGNIZED, pid/lid, QTF. xxxxxxxxxxxx yyyyyyyyyy	FF	The network message received was not recognized.
NETWORK BLOCKS OUT OF SEQUENCE pid/lid, QTF.	F	Blocks sent on the network were out of sequence.

NETWORK MESSAGE BLOCK SIZE ERROR, pid/lid, QTF.	F	A message block on the network is larger than allowed.
NETWORK FULL, QTF.	F	The maximum number of QTF's are already netted on.
pid - CONNECTION LIMIT REACHED QTF.	I	The local number of connections has been reached and one move was tried.
pid - DESTINATION NOT RESPONDING, QTF.	I	The remote NAD is not responding.
pid - PID DISABLED OR INVALID AT DESTINATION, QTF.	I	The PID at the remote end is either invalid or is disabled.
pid - LID DISABLED OR NOT valid, QTF.	I	The LID specified in a connection request is not valid at the local end or is disabled.
pid - NAD(S) full, QTF.	I	The NADs have reached their maximum number of connections.
pid - NETWORK SHUT DOWN IN PROCESS, QTF.	I	The local network has been requested to shut down.
pid - REMOTE APPLICATION FULL OR INVALID NAME, QTF.	I	The name in the request message is either illegal at the destination or already has its limit of connections.
PID/LID FILE QTFTID NOT FOUND QTF.	F	The LID/PID association file was not found.
QFM ERROR READING SYSTEM SECTOR, xxxxxxxx, QTF.	FF	Error in recording the system sector of the file xxxxxxxx.
SENT TO PID xxx, RECEIVED FROM PID yyy, QTF.	I	A file was sent to PID xxx and the remote side said that it was PID yyy.
SUBSYSTEM NOT ACTIVE, QTF.	F	RHFAM was not active.
TOO MANY PIDS/LIDS SPECIFIED IN THE QTFTID FILE, QTF.	F	The LID/PID file specified too many unique lids.
UNEXPECTED QAC ERROR (EC=xx), QTF.	F	An unexpected QAC error code was returned.

```

UNRECOGNIZED COMMAND RECEIVED |FF| A command received on the network
pid/lid, QTF.                  | | was not recognizable.
xxxxxxxxxxxx yyyyyyyyyy       | |
                                | |

```

pid/lid pid is the originating host PID and lid is the LID to which the file was sent

ac Application connection number

pid pid to which a connection request was sent

xxxxxxxxxxxx Header of unrecognized message

yyyyyyyyyy First word of unrecognized message

rc Reason code for FC/Breaks

- 1 - Connection timed out
- 2 - Illegal message received
- 3 - Illegal sequence of commands

error codes describing the reason a file transfer errored out

- 01 Illegal message received on the network
- 02 Illegal sequence of level 7 messages
- 03 Connection timed out
- 04 Invalid con/req/n message
- 05 QFM error in reading the system sector
- 06 User not allowed to use the network or user security count exhausted or invalid user number
- 07 invalid DD parameter
- 08 Forced drop
- 09 Application block number error
- 10 Network block type error
- 11 Unrecognized network block type
- 12 Message received after FC/Break sent is not a CON/CB/N or a CON/END/R
- 13 Error in queueing file on receiver
- 14 Move card error
- 15 Attribute qualifier not S
- 16 Invalid protocol ID
- 17 Output device type error
- 18 Mode of access error
- 19 Invalid data declaration
- 20 Required attribute missing
- 21 Both implicit and explicit texts missing
- 22 Explicit route text and not an explicit route

23 Explicit route with no explicit route text
24 Invalid user name
25 Data declaration not valid for input file
26 Required attribute being ignored
27 Invalid qualifier for parameter
28 Parameter begin modified
29 Required parameter not received
30 File not wanted by receiver
31 Invalid user text
32 Remote side error on transfer
33 RNEG received
34 NETXFR error during data transfer
35 FC/BRK received
36 Connection broken
37 Connection timed out
38 NAK retry limit reached
39 Invalid control statement
40 Keyword not found
41 Invalid separator
42 Invalid parameter
43 Invalid combination of DC/EC parameters

NOS SYSTEM MESSAGES

This section contains a sorted listing of console messages followed by the routine which issued the message and an explanation of the message.

The messages may appear on the following displays:

Job Status (B) Display
System Dayfile (A) Display
Console Display During Deadstart

DSP - INVALID DID/OID. (DSP)

An illegal destination ID or originating MF ID was specified.

DSP - NOT VALIDATED TO USE ST. (DSP)

The user specified a logical ID on a ROUTE call and was not validated to use logical IDs.

DSP - SECURITY COUNTS ARE ZERO. (DSP)

During the routing of a queue file, it was determined that the user's security count is exhausted. The job is aborted.

ILLEGAL LID. (IAJ)

An LID was specified on the job card that is illegal.

JOB NOT VALIDATED for MF. (IAJ)

An LID was specified on the job card by a user not validated to use LIDs.

NO MACHINE ID SPECIFIED. (SET)

No MID directive was encountered in CMRDECK.

QFM INVALID DESTINATION ID SPECIFIED. (QFM)

A submitted file has an illegal LID on the job card.

SFM - CURRENT ATTRIBUTE IS NOT IN LID TABLE. (SFM)

An LID entry was requested to be altered but the table entry differs from the entry being used by the caller.

SFM - LID NOT LEGAL. (SFM)

An attempt was made to alter an LID that does not exist in the LIDT.

SFM - LID TABLE TOO LARGE FOR BUFFER.

An attempt was made to get a copy of the LID table but the LIDT was larger than the caller's buffer.

TOO MANY LID ENTRIES. (RMS)

During deadstart, the LIDT became full while entering the LIDs specified in IPRDECK.

INDEX

AC=x (see QTF operator commands)
Access code 4-2
ACCNT 4-9
Accountability message logging, FIP (see FIP accountability message logging)
Accounting Messages 4-19
Accounting Summary 4-18
Active Job Queues display 3-3
AIP (see Application Interface Program)
APPEND 2-4
Application Block Header Format
 For Input Data Blocks 5-13
 For Output Data Blocks 5-16
Application Block Header Format
 For Input Supervisory Messages 5-19
 For Output Supervisory Messages 5-22
Application Interface Program 5-1
Application Table 4-9
Application Table commands
 DISABLE,xx 4-16
 ENABLE,xx 4-16
Application Table display 4-16
Application echo checking 4-9
Applications Table 4-14,15
ATTACH 2-3,4

Binary Maintenance Log 4-18
BML (see Binary Maintenance Log)

CHANGE 2-4
CHARGE 2-3
CMRDECK 4-3,4
COMQAPR parameters 4-8
COMSNAD parameters 4-8
COMSRHF NADT 4-18
COMSRHF parameters 4-7
CONF macro 4-7
Control commands 3-1
Controlware 4-3,6,8
Correlation message 4-19

Data Block Header Formats 5-13
Data Set 1-4
DC=IX
 For MFQUEUE 2-7
 For explicit routing text 2-8
Deadstart Logical ID table setup 4-4
Deadstart NAD autoloading control 4-3

Dedicated Path Manager Option, RHFAM (see RHFAM Dedicated Path
Manager Option)
DEFINE 2-3,4
Detail Status Logging Option, RHFAM (see RHFAM Detail Status Logging
Option)
DISABLE, RHFAM 3-1
DISABLE,xx (see Application Table commands)
DMPNAD control statement 4-13 (see also X.DMPNAD)

ECHO 4-9
ENABLE, RHFAM 3-1
ENABLE,xx (see Application Table commands)
Explicit routing text 2-8

Facility Interface Program 5-1
FCOPY control statement 2-11
File
 Alteration 2-1
 Maintenance 2-1
 Transfers 2-1
FileName Table display 3-2
File size message 4-19
FIP (see Facility Interface Program)
FIP accountability message logging 4-9
FIPDB 4-9
FIPDB 5-12
FNTLIST 3-3

GET 2-4

H display (see File Name Table display)
Hardware requirements 1-4
Host mainframe 1-1

IDLE command
 DSD 3-2
 QTF K display 3-5
 RHFOU 3-2
Input File Staging Accounting 4-20
Interface functions 1-3
IPRDECK 4-4

K.APPL 4-14
K.LIDT 4-14
K.NADT 4-14

LBC 4-3,6
LCN (see Loosely Coupled Network)
Lines of Text 2-1,3
Linked ID 1-2
Linked mainframe 1-1
LOADBC control statement 4-6 (see also X.LOADBC)
Logical ID Table Command 4-17
Logical ID table 4-4
Logical Identifier Table 4-14,17
Loosely Coupled Network 1-1

Maintenance host 4-18
Maintenance Logging Transfer Facility 4-18
MFLINK control statement 2-1
MFLINK session 2-2
MFQUEUE control statement 2-7
Mid 4-17
MLTF (see Maintenance Logging Transfer Facility)
MMF (see Multimainframe)
MMF ID 1-2
Multimainframe 1-2

NA option 2-3
NAD (see Network Access Device)
NAD Autodump 4-13
NAD autoloading 4-6
NADT (see Network Address Table)
NAT (see Network Address Table command)
NETDBG 5-11
NETGET 5-6
NETGETL 5-7
NETOFF 5-4
NETON 5-2
NETPUT 5-9
NETWAIT 5-5
Network Access Device 1-4
Network Access Device EST entry 4-1
Network Access Device address 4-1
Network Address Table 4-5,7,14
Network Address Table command 4-15
Network Address Table display 4-15
Network Failure Processing 4-22
NETXFR 5-9
n.RHFffff 3-1; 4-4
NTPXFR 5-11

OFF (see QTF operator commands)
OFF=xx (see QTF operator commands)
ON (see QTF operator commands)
ON=xx (see QTF operator commands)
Output File Staging Accounting 4-20

PACKNAM 2-4
Permanent File Staging Accounting 4-21
Permanent File Transfer Facility 2-1
PERMIT 2-5
Physical access code (see Access code)
Physical ID 1-2
Physical network address (see Network Access Device address)
PID (see Physical ID)
PID/LID File 4-12
Procedure file for RHFAM subsystem 4-5
PTF (see Permanent File Transfer Facility)
PURGE 2-5
PURGEALL 3-2

- Q display (see Active job queues display)
QTF (see Queue File Transfer Facility)
QTF K display 3-4
QTF operator commands 3-4
QTFTxx 4-12
Queue File Transfer Facility 2-7

Remote Host Facility v
Remote Host Facility Operator Utility 4-14
REPLACE 2-5
RESET (see QTF operator commands)
RHF (see Remote Host Facility)
RHF Application Installation Options 4-9
RHFAM 1-1
RHFAM Activities 1-2
RHFAM Dedicated Path Manager Option 4-12
RHFAM Detail Status Logging Option 4-13
RHFAM initialization 4-4
RHFAM operator commands 3-1
RHFAM parameters 4-9
RHFAM startup 4-5
RHFAM termination 3-2
RHFDBG 5-1
RHFffff (see n.RHFffff)
RHFGET 5-1
RHFGETL 5-1
RHFIO library 5-1,11
RHFOFF 5-1
RHFON 5-1
RHFOU (see Remote Host Facility Operator Utility)
RHFPUT 5-1
RHFWAIT 5-1
RHFXHR 5-1
ROUTE
 Explicit routing text 2-8
 NOS ROUTE control statement 2-9

SA command (see Logical ID Table command)
SAVE 2-5
Set Attribute command (see Logical ID Table command)
ST parameter
 For MFLINK 2-2
 For MFQUEUE 2-7
 For the NOS job statement 2-10
 For the NOS ROUTE statement 2-9
STOP
 DSD command 3-2
 QTF K display 3-5
Subsystem installation 4-7
Supervisory Block Header Formats 5-19

TCU (see Trunk Control Unit)
Text file
 For MFLINK 2-1,2
 For MFQUEUE 2-8
Trunk Control Unit 1-4
Trunk Control Unit enable mask 4-1,2

USER 2-3

Validation Summary 4-21

X.DMPNAD 4-22 (see also DMPNAD)
X.LOADBC 4-22 (see also LOADBC)

Zid 4-17

COMMENT SHEET

MANUAL TITLE: CDC Remote Host Facility Handbook

PUBLICATION NO.: 60459060

REVISION: 01

NAME: _____

COMPANY: _____

STREET ADDRESS: _____

CITY: _____ STATE: _____ ZIP CODE: _____

This form is not intended to be used as an order blank. Control Data Corporation welcomes your evaluation of this manual. Please indicate any errors, suggested additions or deletions, or general comments below (please include page number references).

Please Reply

No Reply Necessary

CONTROL DATA CORPORATION

NO POSTAGE STAMP NECESSARY IF MAILED IN U.S.A.

FOLD

FOLD



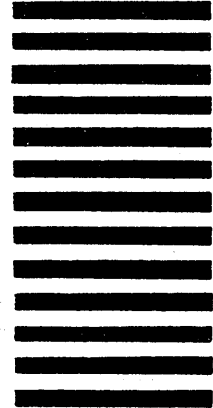
NO POSTAGE
NECESSARY
IF MAILED
IN THE
UNITED STATES

BUSINESS REPLY MAIL
FIRST CLASS PERMIT NO. 8241 MINNEAPOLIS, MINN.

POSTAGE WILL BE PAID BY

CONTROL DATA CORPORATION

Publications and Graphics Division
ARH219
4201 North Lexington Avenue
Saint Paul, Minnesota 55112



FOLD

FOLD

CORPORATE HEADQUARTERS, P.O. BOX 0, MINNEAPOLIS, MINN. 55440
SALES OFFICES AND SERVICE CENTERS IN MAJOR CITIES THROUGHOUT THE WORLD

LITHO IN U.S.A



CONTROL DATA CORPORATION