60459320

CONTROL DATA CORPORATION

NOS VERSION 2 INSTALLATION HANDBOOK

CDC® COMPUTER SYSTEMS:
- CYBER 170
- CYBER 70
  - MODELS 71, 72, 73, 74
- 6000

REVISION RECORD

REVISION	DESCRIPTION
A	Handbook released; reflects NOS 2.0 (which succeeds NOS 1.4) at PSR level 562/552 and supersedes the
(04-26-82)	NOS Version 1 Installation Handbook. It documents the CYBER 170 Computer System, Models 825, 835,
	and 855, and 819 disks; includes a section on APRDECK entries; adds IOB, PPB, DOWN, ISHARE, and
	PRESET CMRDECK entries; drops documentation of ACCOUNT, AUTOROLL, BATCHIO, EI200, MAGNET, MSS
	MASTER, PRIORITY, QPROTECT, TELEX, USERS, and VALID IPRDECK entries; adds BIO, EXTENDED STACK
	PURGING, LOGGING, MAP, MASTER MSS, SECONDARY USER CARDS, and UNLOCK IPRDECK entries; and documents
	the changed format of IPRDECK entries.
	The handbook also documents the two new common disk area utility options (environment interface and
	microcode) and three new decks for NOS installation parameters (COMSMSC, COMSSFS, and COMSSRU). It
	drops documentation of decks COMPCMX, COMSEXP, and RESEQ, CPUMTR segmentation, and procedure file
	LIBMOD.
	The handbook drops documentation of Advanced Access Methods Version 1, ALGOL Version 4, COBOL
	Version 4, COBOL Version 4 to COBOL Version 5 Conversion Aids, CYBER Database Control System
	Version 1, Data Description Language Version 2, Time-Sharing Module, Export/Import Module, and
	TAF/TS Transaction Facility Data Manager. The handbook also drops documentation of the following
	equipment: 841, 65x, 6671/6676/2550-100.
B	Revised to reflect NOS 2.1 at PSR summary level 580/577. This revision documents the CYBER 170
(01-27-83)	Computer System models 815, 865, and 875, and the 885-42 Disk Storage Subsystem; adds the Remote
	Diagnostic Facility and Remote Host Facility; revises installation procedures for NOS 2, Cross, CCP,
	and NAM; drops the use of procedures GENJOB and SYSTEM; adds procedures SETUP and MISCGET; adds
	production environment installation through use of a global library; adds LID, PRIVILEGED RDF, and
	RESIDENT RDF IPRDECK entries; adds short printer paper parameter to unit record EST entry; adds
	two-port multiplexer, network access device, and MAP III equipment EST entries; adds EST entry for
	the 885-42 disk; drops IOB and UEC CMRDECK entries; adds LIDT and XM CMRDECK entries; revises
	release tape contents; and includes editorial and technical corrections. Because of extensive
	changes, change bars and dots are not used. This edition obsoletes all previous editions.
Publication No. 60459320	

REVISION LETTERS I, O, Q, S, X AND Z ARE NOT USED

Address comments concerning this manual to:
Control Data Corporation
Publications and Graphics Division
4201 North Lexington Avenue
St. Paul, Minnesota 55112

or use Comment Sheet in the back of this manual.

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
Front Cover	–
Inside Front Cover	B
Title Page	–
2	B
3	B
4	B
5	B
6	B
7	B
8	B
9	B
10	B
11	B
12	B
13	B
14	B
15	B
16	B
Divider	–
1-1	B
1-2	B
1-3	B
1-4	B
1-5	B
1-6	B
1-7	B
1-8	B
2-1	B
2-2	B
2-3	B
2-4	B
2-5	B
2-6	B
2-7	B
2-8	B
2-9	B
2-10	B
2-11	B
2-12	B
2-13	B
2-14	B
2-15	B
2-16	B
2-17	B
2-18	B
2-19	B
2-20	B
2-21	B
2-22	B
2-23	B
2-24	B
2-25	B
2-26	B
2-27	B
2-28	B
2-29	B
2-30	B
2-31	B
2-32	B
2-33	B
2-34	B
2-35	B
2-36	B
2-37	B
2-38	B
2-39	B
2-40	B
2-41	B
2-42	B
2-43	B
2-44	B
2-45	B
2-46	B
2-47	B
2-48	B
2-49	B
2-50	B
3-1	B
3-2	B
3-3	B
3-4	B
3-5	B
3-6	B
3-7	B
3-8	B
3-9	B
3-10	B
3-11	B
3-12	B
3-13	B
3-14	B
3-15	B
3-16	B
3-17	B
3-18	B
3-19	B
3-20	B
3-21	B
3-22	B
3-23	B
3-24	B
3-25	B
3-26	B
3-27	B
3-28	B
3-29	B
3-30	B
3-31	B
3-32	B
3-33	B
3-34	B
4-1	B
4-2	B
4-3	B
4-4	B
4-5	B
4-6	B
4-7	B
Divider	–
5-1	B
5-2	B
5-3	B
5-4	B
5-5	B
5-6	B
5-7	B
5-8	B
5-9	B
5-10	B
5-11	B
5-12	B
5-13	B
5-14	B
5-15	B
5-16	B
5-17	B
5-18	B
5-19	B
5-20	B
5-21	B
5-22	B
5-23	B
5-24	B
5-25	B
5-26	B
5-27	B
5-28	B
5-29	B
5-30	B
5-31	B
5-32	B
5-33	B
5-34	B
5-35	B
5-36	B
5-37	B
5-38	B
5-39	B
5-40	B
5-41	B
5-42	B
5-43	B
5-44	B
5-45	B
5-46	B
5-47	B
5-48	B
5-49	B
5-50	B
5-51	B
5-52	B
5-53	B
5-54	B
5-55	B
5-56	B
5-57	B
5-58	B
5-59	B
5-60	B
5-61	B
5-62	B
5-63	B
5-64	B
5-65	B
5-66	B
5-67	B
5-68	B
5-69	B
5-70	B
5-71	B
5-72	B
5-73	B
5-74	B
5-75	B
5-76	B
5-77	B
5-78	B
6-1	B
6-2	B
6-3	B
6-4	B
6-5	B
6-6	B
6-7	B
6-8	B
6-9	B
6-10	B
6-11	B
6-12	B
6-13	B
6-14	B
6-15	B
Divider	–
7-1	B
7-2	B
7-3	B
7-4	B
7-5	B
7-6	B
7-7	B
7-8	B
7-9	B
7-10	B
7-11	B
7-12	B
7-13	B
7-14	B
7-15	B
7-16	B
7-17	B
7-18	B
7-19	B
7-20	B
7-21	B
7-22	B
7-23	B
7-24	B
7-25	B
7-26	B
7-27	B
7-28	B
7-29	B
7-30	B
7-31	B
7-32	B
7-33	B
7-34	B
7-35	B
7-36	B
7-37	B
7-38	B
7-39	B
7-40	B
7-41	B
7-42	B
7-43	B
7-44	B
7-45	B
7-46	B
7-47	B
7-48	B
7-49	B
7-50	B
7-51	B
7-52	B
7-53	B
7-54	B
7-55	B
7-56	B
7-57	B
7-58	B
8-1	B
8-2	B
8-3	B
8-4	B
9-1	B
9-2	B
9-3	B
9-4	B
9-5	B
9-6	B
9-7	B
9-8	B
9-9	B
9-10	B
9-11	B
9-12	B
9-13	B
9-14	B
9-15	B
9-16	B
9-17	B
9-18	B
9-19	B

PAGE	REV
9-20	B
9-21	B
9-22	B
9-23	B
9-24	B
9-25	B
9-26	B
9-27	B
10-1	B
10-2	B
Divider	-
11-1	B
11-2	B
11-3	B
11-4	B
11-5	B
11-6	B
11-7	B
11-8	B
11-9	B
11-10	B
Divider	-
A-1	B
A-2	B
A-3	B
A-4	B
B-1	B
B-2	B
B-3	B
C-1	B
C-2	B
C-3	B
C-4	B
C-5	B
C-6	B
C-7	B
C-8	B
C-9	B
C-10	B
C-11	B
C-12	B
C-13	B
C-14	B
C-15	B
C-16	B
C-17	B
C-18	B
C-19	B
C-20	B
C-21	B
C-22	B
C-23	B
D-1	B
D-2	B
D-3	B
D-4	B
D-5	B
D-6	B
Index-1	B
Index-2	B
Index-3	B
Index-4	B
Index-5	B
Index-6	B
Index-7	B
Index-8	B
Index-9	B
Index-10	B
Index-11	B
Index-12	B
Index-13	B
Index-14	B
Comment Sheet	B
Back Cover	-

PREFACE

This handbook describes the installation of the CONTROL DATA® Network Operating System (NOS) Version 2.1 and its products. NOS controls the operation of CDC® CYBER 170 Computer Systems; CDC CYBER 70 Computer Systems, Models 71, 72, 73, and 74; and CDC 6000 Computer Systems.

AUDIENCE

This handbook is written for the systems analyst who is familiar with the COMPASS assembly language, the SYMPL programming language, the Update and Modify source file maintenance utilities, the hardware configuration on which NOS is installed, and the NOS commands.

ORGANIZATION

Tabbed pages divide the handbook into five parts. The first part, Installation Process, is four sections describing the installation process. Section 1 describes the minimum hardware requirements for NOS 2.1, provides an overview of the entire installation process, and lists the operating system and optional products. Section 2 describes the installation procedures for the NOS products, except for the CYBER Cross System (Cross) and Communications Control Program (CCP). The installation procedures for Cross and CCP compose section 3. Section 4 describes the installation procedures for new controlware.

The second part, Product Modifications, is two sections. Section 5 describes the installation parameters, the directives, and the variations in procedures that are unique to a product. The products are listed in alphabetical order by their installation procedure names. The installation parameters in the NOS decks compose section 6.

The titles of the third (CMRDECK, APRDECK, IPRDECK, LIBDECK), fourth (BINEDIT and BNP), and fifth (Appendixes) parts are self-explanatory.

CONVENTIONS

For NOS, the words command and control statement are interchangeable.

Extended memory for model 176 is large central memory extended (LCME). Extended memory for models 815, 825, 835, 855, 865, and 875 is unified extended memory (UEM). Extended memory for models 865 and 875 may also include extended core storage (ECS) or extended semiconductor memory (ESM). Extended memory for all other NOS computer systems is either ECS or ESM. ECS and ESM are the only forms of extended memory that can be shared in an ECS multimainframe complex and can be accessed by a distributive data path (DDP).

In this manual, ECS refers to both ECS and ESM, and extended memory refers to all forms of extended memory unless otherwise noted. However, when referencing extended memory in the context of an ECS multimainframe complex or DDP access, UEM and LCME are excluded.

Throughout this handbook, a cross-reference to either the NOS Version 2 Reference Set, Volume 3 System Commands, or to the NOS Version 2 Reference Set, Volume 4 Program Interface is in the form: refer to the NOS 2 Reference Set, Volume n (n=3 or 4).

You must enter uppercase letters within command formats exactly as given; replace lowercase letters with appropriate characters as described in the text.

Installation jobs can be submitted as batch jobs, called interactively to be run as batch jobs, or run at the console under DIS. In this handbook all installation job samples are batch jobs and have the following form (documentation for the following three commands is in the NOS 2 Reference Set, Volume 3).

```
job command.
USER,username,password,familyname.
CHARGE,*.
  .
  .
  .
```

At all sites, you must specify job and USER commands. The CHARGE command is not required at all sites.

In this handbook, press CR means you are to press the carriage return key on the console.

In this handbook, the term CYBER 70 Computer Systems refers to models 71, 72, 73, and 74 only.

RELATED PUBLICATIONS

Control Data manuals are available through Control Data sales offices or Control Data Literature and Distribution Services (308 North Dale Street, St. Paul, Minnesota 55103).

The NOS Version 2 Manual Abstracts (publication number 60485500) is a pocket-sized manual containing brief descriptions of the contents and intended audience of all NOS 2 manuals and optional product manuals.

Control Data also publishes a Software Publications Release History (publication number 60481000) of all software manuals and revision packets it has issued. This history lists the revision level of a particular manual that corresponds to the level of software installed at the site.

Programming information for the various forms of extended memory can be found in the COMPASS Reference Manual and in the appropriate computer system hardware reference manual. Hardware descriptions of the various forms of extended memory can be found in the following manuals.

Control Data Publication	Publication Number
Extended Semiconductor Memory Hardware Reference Manual	60455990
Extended Core Storage Reference Manual	60347100
Extended Core Storage II and Distributive Data Path Reference Manual	60430000

The following is a list of NOS 2 manuals, optional product manuals, and related hardware manuals.

NOS 2 MANUALS

Control Data Publication	Publication Number
Common Memory Manager Version 1 Reference Manual	60499200
COMPASS Version 3 Reference Manual	60492600
CYBER Loader Version 1 Reference Manual	60429800
CYBER Record Manager Advanced Access Methods Version 2 Reference Manual	60499300
CYBER Record Manager Basic Access Methods Version 1 Reference Manual	60495700
FORM Version 1 Reference Manual	60496200
Modify Version 1 Reference Manual	60450100
NOS Version 2 Applications Programmer's Instant	60459360
NOS Version 2 Diagnostic Index	60459390
NOS Version 2 Network Terminal User's Instant	60459380
NOS Version 2 Operator/Analyst Handbook	60459310
NOS Version 2 Reference Set, Volume 3 Systems Commands	60459680
NOS Version 2 Reference Set, Volume 4 Program Interface	60459690
NOS Version 2 System Maintenance Reference Manual	60459300
NOS Version 2 Systems Programmer's Instant	60459370
On-Line Maintenance Software Reference Manual	60454200
SYMPL Version 1 Reference Manual	60496400

OPTIONAL PRODUCT MANUALS

Control Data Publication	Publication Number
ALGOL-60 Version 5 Reference Manual	60481600
APL Version 2 Reference Manual	60454000
BASIC Version 3 Reference Manual	19983900
COBOL Version 5 Reference Manual	60497100

Control Data Publication	Publication Number
Concurrent Maintenance Library Reference Manual	60455980
Conversion Aids System Version 3 Reference Manual	19265358
CYBER Cross System Version 1 Build Utilities Reference Manual	60471200
CYBER Cross System Version 1 Macro Assembler Reference Manual	96836500
CYBER Cross System Version 1 Micro Assembler Reference Manual	96836400
CYBER Cross System Version 1 PASCAL Reference Manual	96836100
CYBER Interactive Debug Version 1 Reference Manual	60481400
Data Catalogue 2 Reference Manual	60483200
DMS-170 CYBER Database Control System Version 2 Data Administration Reference Manual	60485200
DMS-170 CYBER Database Control System Version 2 Application Programming Reference Manual	60485300
DMS-170 Query Update/CYBER Record Manager Data Administration Reference Manual	60482100
FORTRAN Data Base Facility Version 1 Reference Manual	60482200
FORTRAN Extended Version 4 Common Library Mathematical Routines Reference Manual	60498200
FORTRAN Extended Version 4 Reference Manual	60497800
FORTRAN Extended Version 4 to FORTRAN Version 5 Conversion Aids Program Version 1 Reference Manual	60483000
FORTRAN Version 5 Common Library Mathematical Routines Reference Manual	60483100
FORTRAN Version 5 Reference Manual	60481300
Message Control System Version 1 Reference Manual	60480300
NAM Version 1/CCP Version 3 Reference Manual	60499500
Network Definition Language Version 1 Reference Manual	60480000
NOS Version 2 Applications Installation Handbook	84002760
PASCAL 1 Reference Manual	60485600
PL/I Version 1 Reference Manual	60388100
Query Update Version 3 Reference Manual	60498300
Remote Batch Facility Version 1 Reference Manual	60499600

Control Data Publication	Publication Number
SIMULA 1 Reference Manual	60234800
Sort/Merge Versions 4 and 1 Instant Manual	60497600
Sort/Merge Versions 4 and 1 Reference Manual	60497500
Sort/Merge Version 5 Reference Manual	60484800
TAF Version 1 Reference Manual	60459500
TAF/CRM Data Manager Version 1 Reference Manual	60459510
Text Editor Version 1 Reference Manual	60436100
Update Version 1 Reference Manual	60449900
XEDIT Version 3 Reference Manual	60455730
8-Bit Subroutines Version 1 Reference Manual	60495500

HARDWARE MANUALS

Control Data Publication	Publication Number
CYBER 70 Computer System Model 71 Hardware Reference Manual	60453300
CYBER 70 Computer System Model 72 Hardware Reference Manual	60347000
CYBER 70 Computer System Model 73 Hardware Reference Manual	60347200
CYBER 70 Computer System Model 74 Hardware Reference Manual	60347400
CYBER 170 Computer System Hardware Reference Manual	60420000
CYBER 170 Computer Systems Models 720, 730, 740, 750, 760, and 176 (Level B/C) Hardware Reference Manual	60456100
CYBER 170 Computer System Model 825 Hardware Reference Manual	60469350
CYBER 170 Computer Systems Model 835 and 855 Hardware Reference Manual	60469290
CYBER 170 Computer Systems Models 865 and 875 Hardware Reference Manual	60458920

Control Data Publication	Publication Number
6400/6500/6600 Computer Systems Reference Manual	60100000
2550-2 Host Communication Processor Hardware Reference Manual	74375500

DISCLAIMER

NOS and its product set are intended to be used only as described in this document. Control Data cannot be responsible for the proper functioning of undescribed features or parameters.

CONTENTS

APPENDIXES

INDEX

FIGURES

TABLES

OVERVIEW 1

HARDWARE REQUIREMENTS

The minimum hardware requirements for installing NOS 2.1 are:

- One CYBER 170, CYBER 70, or 6000 Computer System.
- 48K (K represents 1024) words of memory or, if your site is using network products, 96K words of memory.
- One CC545 display console.
- Ten peripheral processors (PPs).
- One 844 or 885-11/12 disk unit.
- Either one 580 line printer and a 405 card reader, or one 734-CYBER 18 terminal that is physically located with the central computer and is driven by a 255x communications subsystem.
- Either one disk controller and one tape controller or one 7152 controller.
- Two 667, 669, 677, or 679 tape units.

Some of the products have additional hardware requirements. They are listed in section 5.

INSTALLATION PROCESS OVERVIEW

The following steps outline the procedures necessary to install the operating system and products. These steps are fully explained in section 2.

NOTE

You can install the NOS system described in this manual on a dedicated system by using the unconfigured deadstart tape (released version) or in a production environment running an earlier release of NOS 2.

STEP 1	Deadstart using the released version of the deadstart tape.
STEP 2	Create or modify the CMRDECKs, APRDECKs, IPRDECKs, and LIBDECKs as necessary.
STEP 3	Create a new deadstart tape. You must also create a deadstart tape when the installation process is complete. Redeadstart with the new deadstart tape.

STEP 4	Install the common testing and initialization (CTI) module from the released version of the deadstart tape on a disk. This step is optional, except it is mandatory for models 815, 825, 835, and 855.
STEP 5	Install the CTI/MSL (maintenance software library) disk area utility on a disk. This step is optional, except it is mandatory for models 815, 825, 835, and 855.
STEP 6	Install the hardware initialization and verification software (HIVS) module on a disk. This step is optional, except it is mandatory for models 815, 825, 835, and 855.
STEP 7	Create the validation and project profile files, as necessary.
STEP 8	Generate the permanent files used in the installation process. You can either enter the jobs that generate the permanent files as batch jobs, call the jobs interactively, or enter them from the console under DIS.†
STEP 9	Determine which installation parameter settings require changes and which site, suggested, and corrective codes need to be added to the operating system and products. Make the changes in the correct format (Modify or Update) and put them on file USER, which will be accessed during the installation procedure. CCP and CROSS (section 3) use files UCCP and UCRS.
STEP 10	Install the operating system and products.
STEP 11	Check the installation of the products by running the verification jobs.
STEP 12	Install new controlware, if any.
STEP 13	Put the product files in the appropriate permanent file catalogs as required.

OPERATING SYSTEM AND OPTIONAL PRODUCTS

Lists of the contents of the base operating system and the optional products available under NOS 2.1 follow.

The base operating system includes the following CDC products.

BINEDIT

CEDIAG 1.2

Common Memory Manager Version 1.1

COMPASS Version 3.6

CTI

CYBER Common Utilities

CYBER Control Language Version 1.0

CYBER Loader Version 1.5

CYBER Record Manager

  Basic Access Methods Version 1.5

  Advanced Access Methods Version 2.1

FORM Version 1.1

HIVS

Modify Version 1.2

Network Operating System Version 2.1

Product Texts

†For each installation job run under DIS, you must initiate a new DIS session.

Product Texts I/O

Text Editor Version 1.4

Update Version 1.4

8-Bit Subroutine Version 1.1

The maintenance package, ordered separately, contains the following CDC products.

CYBRLOG

Maintenance Tools

SYMPL 1.4

881/883 Pack Formatting

The following CDC products are optional and are ordered separately.

ALGOL-60 Version 5.1

APL Version 2.1

BASIC Version 3.5

COBOL Version 5.3

Common Code Generator Version 1.0

Communications Control Program Version 3.5

Conversion Aids System Version 3.0

CYBER Cross System Version 1.2

CYBER Database Control System Version 2.3

CYBER Interactive Debug Version 1.1

CYBER Mass Storage Subsystem Version 1.0

Database Utilities Version 1.2

Data Catalogue Version 2.1

Data Description Language Version 3.2

FORTRAN Common Library Version 4.8 with Postmortem Dump Utility

FORTRAN Common Library Version 5.1 with Postmortem Dump Utility

FORTRAN Data Base Facility Version 1.3

FORTRAN Extended Version 4.8

FORTRAN Extended Version 4.8 with Interactive Option

FORTRAN Version 5.1

FORTRAN 4 to 5 Conversion Aid Version 1.0

High Speed I/O Option Version 1.0

Information Management Facility Version 1.1

Interactive Facility Version 1.0

Message Control System Version 1.0

Multimainframe Module Version 1.0

Network Access Method Version 1.5

PASCAL 170 Version 1.0

PL/I Version 1.0

Query Update Version 3.4

Remote Diagnostic Facility

Remote Batch Facility Version 1.5

Remote Host Facility Version 2.0

Remote Host Products 1.0

Sort/Merge Version 4.6

Sort/Merge Version 5.0

Tracer Version 1.0

Transaction Facility Version 1.2

XEDIT Version 3.1

Additional application products are also available with NOS. Refer to the NOS 2 Application Installation Handbook for information on those products and their release materials.

SYSTEM ORGANIZATION NOTES

To learn more about the Modify- and Update-formatted products, examine their decks. You can examine the decks of Modify-formatted products only after executing the COMBINE procedure (refer to section 2). The COMBINE procedure produces a file called RELSOPL residing on either a disk or tape (with VSN=RELSOPL). To examine the decks of Update-formatted products, get a copy of the desired program library from a release tape. Most release tapes contain only one product; the first file is usually the program library. Some release tapes contain several products. For these tapes the program libraries are usually on the first, fifth, ninth, and so on, files. Be sure you get the appropriate file. Refer to appendix C for a description of the release tapes.

Use of the COMMON,SYSTEM command where it is shown in some of the examples in this manual will not produce the intended results if you are running jobs on a NOS 2.0 system. You should either attach the NOS 2.1 deadstart file or issue a tape request for the NOS 2.1 deadstart tape. If you are running on a NOS 2.1 system, use of the COMMON,SYSTEM command is acceptable, provided the user name is validated to use library files.

MODIFY-FORMATTED PRODUCTS

To determine which decks and modification sets are on a Modify-formatted program library, run a job similar to the following:

Job	Comment
job command. USER,username,password,familyname. CHARGE,*. . . . CATALOG,lfn,R. --eoi--	Enter commands that get the desired Modify-formatted program library and give it a local name (lfn).

The R parameter of the CATALOG command rewinds the program library before and after execution of the CATALOG command.

The CATALOG output contains a column labeled TYPE. This column indicates whether the deck is a common deck (OPLC) or not (OPL), and whether its character set is a 63-character or a 64-character set.

Modification sets, if any exist, are listed beneath each deck (record). The names of the modification sets that have been yanked are enclosed in parentheses.

To get documentation of NOS programs from the system program library file, RELSOPL, use Modify (refer to the Modify Reference Manual) and DOCMENT (refer to the NOS 2 Reference Set, Volume 3).

RELSOPL must reside on mass storage. Enter the following sequence of commands from the system console to copy RELSOPL from tape (the default for the COMBINE procedure) onto mass storage. If RELSOPL is placed on a permanent file, you must control access to it as required by the site's license agreement. (The job display program DIS is described in the NOS 2 Operator/Analyst Handbook.)

Entries	Comments
X.DIS.	
SUI,377776.	User index for user name LIBRARY.
FAMILY,familyname.	
VSN,TAPE=RELSOPL.	
LABEL,TAPE,D=PE.	Adjust density parameter as needed.
PURGE,OPL/NA.	
DEFINE,OPL.	
COPYBF,TAPE,OPL.	
PERMIT,OPL,username=m.	Control access with the PERMIT command. Specify the access mode, m, for the user name.

If you want a catalog of the system deadstart file and RELSOPL, run the following job.

Job	Comments
job command.	
USER,username,password,familyname.	The user name must be validated to use library files.
CHARGE,*.	
COMMON,SYSTEM.	
CATALOG,SYSTEM,N.	
ATTACH,OPL/UN=LIBRARY.	This job assumes that RELSOPL was saved as a direct access file named OPL under the user name LIBRARY.
CATALOG,OPL,N.	
--eoi--	

In most cases, the OPL deck name, which is the name required on an *EDIT directive for Modify, is the same as the program name on the deadstart tape. For most overlays, such as 9AA, 9AB, and so forth, the deck in which they are contained is shown in the Comments area of a CATALOG listing of the system deadstart file. For instance, for 9AA the comment reads:

DSD - DISPLAY A - DAYFILE MESSAGES.

Therefore, to obtain a listing of overlay 9AA you must use *EDIT DSD.

To list the active lines with sequencing information in a deck on a Modify program library, run a job similar to the following:

Job	Comments
job command.	
USER,username,password,familyname.	
CHARGE,*.	
ATTACH,OPL/UN=LIBRARY.	User must have permission to access OPL. This job assumes that RELSOPL is saved as a direct access file named OPL under the user name LIBRARY.
MODIFY,LO=A,C=0,Z./*EDIT,deck	deck is the name of a deck listed in the preceding job.
--eoi--	

The following examples use Modify to obtain information and documentation of the system.

Example 1:

An assembly listing of the peripheral and central processor communication definitions and macros can be very useful to a person seeking detailed information about NOS. To obtain this information, assemble NOSTEXT. NOSTEXT contains the PP systems communications definitions and the central program communications macro definitions.

The PP systems communications definitions (PPCOM) contain the following information: system constants, PP memory location assignments, PP resident and mass storage driver entry points, monitor function values, the contents of central memory resident (CMR) locations, and so forth.

The central program communications macro definitions (CPCOM) contain the following information: the contents of the job communication area, system request macros, general purpose macros, macros for creation of file environment tables (FETs), and macros to control local file action.

To obtain an assembly listing, enter a job similar to the following:

Job	Comment
job command.	
USER,username,password,familyname.	
CHARGE,*.	
ATTACH,OPL/UN=LIBRARY.	User must have permission to access OPL. This job assumes that RELSOPL is saved as a direct access file named OPL under the user name LIBRARY.
MODIFY,A,Z./*EDIT,NOSTEXT	If the assembly aborts, and you do not want a listing, include the A parameter on the COMPASS command.
COMPASS,I,G=0,B=0.	
--eoi--	

Example 2:

Enter a job similar to the following to obtain the external documentation (comment sections) of a Modify-formatted deck.

Job	Comments
job command.	
USER,username,password,familyname.	
CHARGE,*.	
ATTACH,OPL/UN=LIBRARY.	User must have permission to access OPL. This job assumes that RELSOPL is saved as a direct access file named OPL under the user name LIBRARY.
MODIFY,LO=E,Z./*EDIT,deck	deck is the name of the desired Modify deck.
DOCMENT.	DOCMENT reads the COMPILE file (from Modify) and generates the listing of the external documentation.
--eoi--	

For example, to obtain the documentation for the job display program, enter DIS for the deck option; to obtain the documentation for the magnetic tape executive program, enter MAGNET for deck.

To get a source listing of a Modify-formatted deck, replace the DOCMENT command in the preceding job with the COPYSBF(COMPILE) command.

Example 3:

Use the following job to obtain an assembly listing of the source file maintenance utility Modify and to direct the output to a central site line printer.

Job	Comments
job command.	
USER,username,password,familyname.	
CHARGE,*.	
ATTACH,OPL/UN=LIBRARY.	User must have permission to access OPL.
MODIFY,A,Z,L=LIST,LO=E./*EDIT MODIFY.	
COMPASS,A,I,S=NOSTEXT,B=0,L=LIST.	
ROUTE,LIST,DC=PR.	Direct the output file to a line printer.
--eoi--	

Refer to the Modify Reference Manual for further information on listing various programs. The following parameters are especially helpful.

Parameter	Description
A	Specifies that the COMPILE file is in compressed format.
Z	Specifies that the Modify directives follow the command terminator. The first character following the terminator is the separator character for all directives on the command. Any 6-bit display code character that is not used in any of the directives can be used as the separator character (including a space). The last directive must not be followed by a terminator character. If you use a terminator, it is read as part of the directive and an error message is issued.

UPDATE-FORMATTED PRODUCTS

To determine which decks and correction identifiers are on an Update-formatted program library, run a job similar to the following:

Job	Comments
job command.	
USER,username,password,familyname.	
CHARGE,*.	
.	Put commands here that get the desired
.	Update-formatted program library and give it
.	a local name of OLDPL.
NOTE,IN./*COMPILE,HISTORY	
UPDATE,I=IN,Q.	
--eoi--	

To determine which decks are common decks on an Update-formatted program library, run a job similar to the following. This job also lists the deck and correction identifiers.

Job	Comments
job command.	
USER,username,password,familyname.	
CHARGE,*.	
.	Put commands here that get the desired
.	Update-formatted program library and give it
.	a local name of OLDPL.
ASSIGN,NE,COMPILE.	
UPDATE,F.	
--eoi--	

To get a list of the active lines with sequencing information of an Update-formatted deck, enter a job similar to the following:

Job	Comments
job command.	
USER,username,password,familyname.	
CHARGE,*.	
.	Put commands here that get the desired
.	Update-formatted program library and give it
.	a local name of OLDPL.
NOTE,IN./*COMPILE,deck	deck is the name of the desired Update-formatted deck.
UPDATE,I=IN,Q,L=7,C=0.	
--eoi--	

INSTALLATION PROCEDURE 2

The installation procedure presented in this section is an expansion of the Installation Process Overview in section 1. The steps correspond to each of the steps in the overview. An additional two subsections describe some of the files used in the installation process and list the user libraries created by the installation process.

Before proceeding with an installation:

- Check tables 2-2 through 2-4 and figure 2-5 to determine if the products you are installing are dependent upon the installation of other products. If so, be sure to execute all the procedures upon which the products depend.

- Make a copy of each release tape to prevent writing over your only copy. One way to do this is to write each tape to a direct access file with the same file name as the tape's VSN. Then use PFDUMP to dump the files to tape.

- Carefully review your plans for modifying the operating system and products. Modifications include site code, CDC-suggested code and corrective code, and product installation parameter changes. CDC-suggested code and corrective code, if any, reside on RELO (refer to appendix C). Refer to the Product Modifications part for the description of product installation parameters and for additional installation information necessary for some products.

All examples in this manual are shown as batch jobs using the release tapes. If you plan to install with the files residing on disk rather than on tape, be aware of the following:

- The product installation procedures use the file name TAPE for the input tape.† If TAPE is not a local file, the installation procedure requests this file with a LABEL command.

- The product installation procedures use the file name RELTAPE for the output tape.†† If you want the final program library on a permanent file, a local file named RELTAPE must exist. It is your responsibility to make RELTAPE a permanent file.

- If the installation job accesses additional tapes, the files residing on disk must have local file names that are the same as the tapes' VSNs. For example, if the tape VSN is REL2B, then the local file name must also be REL2B.

- You can select disk installation by changing the default for the formal parameter DISKINS to YES. Refer to COMDECKS on a listing of DECKOPL. Disk installation requires preloading all release tapes to disk. You must observe certain file naming conventions to control this process properly. You may optionally store the release tapes on one to six removable disk packs. Refer to COMDECKS and user utilities on the listing of DECKOPL for complete instructions.

SEE ERRATA

†For the COMBINE, CROSS, and CCP procedures, the local file name for the input file is the same as the tape's VSN.

††For the Cross and CCP procedures, NEW13x is the file name of the output tape. The VSN of the tape is REL13x. The installation of Cross and CCP is described in section 3.

The following sample job installs FTN5 from permanent files. As previously specified, TAPE is the input file and RELTAPE is the output file. The additionally required tapes, VSN=REL14B and VSN=REL3A, were copied to permanent files REL14B and REL3A prior to this job. Refer to the preceding paragraph if you prefer to have DECKOPL manage the permanent files containing the REL tapes.

Job	Comments
job command.	
USER,username,password,familyname.	
CHARGE,*.	
ATTACH,RELTAPE=REL4E/M=W.	Permanent file REL4E contains the updated FTN5 program library.
COPYEI,RELTAPE,TAPE.	
REWIND,TAPE,RELTAPE.	
ATTACH,REL14B.	
ATTACH,REL3A.	
GET,USER=pfn.	Local modifications for FTN5, if any, must be on file USER.
BEGIN,FTN5,INSTALL,PC=YES.	
--eoi--	

STEP 1 DEADSTART

Deadstart using the release version of the deadstart tape (refer to the NOS 2 Operator/Analyst Handbook for deadstart procedures).

For models 815, 825, 835, and 855 the first time the mainframe is deadstarted or when the CTI, MSL, or HIVS package has changed since being installed, steps 4, 5, and 6 must be run before steps 1, 2, and 3. If in doubt, install CTI, MSL, and/or HIVS by running steps 4, 5, and 6 before steps 1, 2, and 3.

If you require installation procedures before progressing to step 8 in the installation, you can run the job that includes the command BEGIN,,TAPE described in step 8 to set up the INSTALL file.

STEP 2 MODIFY CMRDECKS, APRDECKS, IPRDECKS, AND LIBDECKS

Create or modify the CMRDECKs, APRDECKs, IPRDECKs, and LIBDECKs to fit your site's configuration and put them on a file as text records. CMRDECK modifications are described in section 7, APRDECK modifications are described in section 8, IPRDECK modifications are described in section 9, and LIBDECK modifications are described in section 10 (refer to the NOS 2 System Maintenance Reference Manual for SYSEDIT directives acceptable in LIBDECK).

STEP 3 CREATE DEADSTART TAPE

During the installation process, you may occasionally create a new deadstart tape by using the GENSYS procedure. The installation files must be created (refer to step 8) before creating a new deadstart file. Each time a deadstart tape is made, you may then run the procedure SEED to reduce the file space required by the file PRODUCT. The initial use of GENSYS requires the released deadstart tape as input. Each subsequent use of GENSYS must use as input the last deadstart tape generated by GENSYS.

The format of the GENSYS call is:

```
BEGIN,GENSYS,INSTALL,SYSTEM=ods,NEW=nds,D=density.
```

Parameter	Description
SYSTEM=ods	Local file name for the old deadstart file. If file ods is preassigned, it becomes the base deadstart file. Otherwise a tape label request is issued with VSN=ODS. The default file name is SYSTEM.
NEW=nds	Local file name for the new deadstart file. If file nds is preassigned, the new deadstart file is written on it. Otherwise a tape label request is issued with VSN=NDS. The default file name is NEW.
D=density	Tape density option. If this parameter is omitted, D=PE is assumed. The option applies to ods and nds.

density	Significance
HI	556 cpi (seven-track).
HY	800 cpi (nine-track).
HD	800 cpi (nine-track).
PE	1600 cpi (nine-track).
GE	6250 cpi (nine-track).

Run a job similar to the following to add site-provided binaries, CMRDECKs, APRDECKs, IPRDECKs, and LIBDECKs to the new deadstart tape. Create file USERD so it contains the LIBEDIT directives (refer to NOS 2 Reference Set, Volume 3) to add the modified CMRDECKs, APRDECKs, IPRDECKs, LIBDECKs, and site-provided binaries to the deadstart tape.

Job	Comments
job command.	
USER,username,password,familyname.	
CHARGE,*.	
GET,USERD.	USERD contains the LIBEDIT directives.
GET,lfn=pfn.	lfn (permanent file name is pfn) contains the modified CMRDECKs, APRDECKs, IPRDECKs, and LIBDECKs.
BEGIN,GENSYS,INSTALL,SYSTEM=ods,NEW=nds,D=density.	Parameters are not required if you use the system default.
--eoi--	

STEP 4 INSTALL CTI MODULE

Installing the CTI module on a disk allows deadstarting from that disk; it is required on models 815, 825, 835, and 855 or if the hardware installation verification software or the Maintenance Software Library will be installed at your site.†

Use the following procedure to prepare a scratch disk and install CTI on that disk. If the disk that receives the CTI module is already a system disk, installing CTI leaves the operating system intact.

1. Ensure that the tape controlware (for 66x subsystems) and disk controlware are present and functioning properly (refer to coldstart in the NOS 2 Operator/Analyst Handbook).

2. If the disk receiving the CTI module does not already have a CTI module, ensure that the disk contains no dayfiles, queue, or permanent files that must be preserved.

3. Mount the operating system deadstart tape without the write enable ring and ready the unit.

4. Set the deadstart panel for warmstart from tape; for models 815 or 825, enter the values for warmstart from tape through the deadstart display (refer to the NOS 2 Operator/Analyst Handbook).

5. Press the deadstart switch. The initial options (A) display appears (refer to figure 2-1).

6. Select the U option. The utilities (U) display appears (refer to figure 2-2).

7. Select the I option. Another display appears (refer to figure 2-3). If MSL resides on the disk, go to step 12 of this procedure.

8. Press R. The system requests the channel and equipment numbers of the disk controller and the unit number of the disk that will receive the CTI module.

9. Enter the channel, equipment, and unit numbers for the device and press CR. The following message appears.

```
ENTRY OF (CR) WILL CAUSE
RELEASE OF CTI-MSL/HIVS RESERVED
DISK SPACE
```

10. Press CR. If a previously installed CTI module resides on the disk, the system releases it. If no CTI module resides on the disk, the system prepares the disk to receive the module. If the operation was successful, the following message appears.

```
RELEASE COMPLETE
(CR) TO PROCESS DIFFERENT DEVICE
```

11. Press CR. The I option display appears (refer to figure 2-3).

---

†The off-line MSL is provided only to the customer engineer who is assigned responsibility for a site under the contractual requirements of the Control Data maintenance services agreement. Contact your customer engineer for more information.

```
                    *A*

         (CR) - OS LOAD AUTOMATIC

          O   - DEADSTART WITH OPERATOR
                INTERVENTION

          U   - UTILITIES

          M   - OFFLINE MAINTENANCE†

          I   - INITIALIZE MAINFRAME††

         CTI    vvv†††
```

† When deadstarting from disk, this line appears only if MSL resides on the deadstart disk.

†† This line appears only when deadstarting on a model 815, 825, 835, or 855.

††† vvv represents the version of CTI.

Figure 2-1. CTI Initial Options (A) Display

```
                    *U*

         (BS) - RETURN TO *A* DISPLAY

           C  - CTI/MSL DISK AREA UTILITY

           E  - EXPRESS DEADSTART DUMP

           I  - INSTALL CTI ON RMS

           P  - PRINTER DUMP

           S  - ALTERNATE DEADSTART

           T  - INSTALL MSL/HIVS TO RMS
```

Figure 2-2. Utilities (U) Display

```
ENTER ONE OF THE FOLLOWING

(CR) - INSTALL DEADSTART
       MODULE ON DISK

  R  - RELEASE CTI-MSL/HIVS RESERVED
       DISK SPACE
```

Figure 2-3. I Option Display

12. Press CR. The following warning message appears.

```
        *WARNING*

PERMANENT FILES MAY BE LOST
IF CTI IS NOT ALREADY
INSTALLED ON THIS DEVICE

(CR) TO CONTINUE
```

13. Press CR. The following message appears.

    ```
    INSTALL DISK DEADSTART MODULE
    ```

    The system then requests the channel and equipment numbers of the disk controller and the unit number of the disk that will receive the CTI module.

14. Enter the channel, equipment, and unit numbers. The message

    ```
    INSTALLING CTI TO DISK
    ```

    appears during the installation process.

    If CTI is loaded successfully, the following message appears.

    ```
    INSTALL COMPLETE
    (CR) TO PROCESS DIFFERENT DEVICE
    ```

15. If your site has more than one system disk, press CR and repeat steps 8 through 14 of this procedure to install CTI on each of the system disks.

16. Press the deadstart switch to return to the A display. The current version of CTI appears at the bottom of the display. The CTI installation process is complete. The disk on which CTI resides must be properly formatted and must have utility flaw maps present.

STEP 5 INSTALL CTI/MSL DISK AREA UTILITY

The CTI/MSL disk area (CDA) utility establishes a common disk area, in which programs and data are stored for use by CTI and MSL† routines, and allows you to modify the CTI default parameters. The CDA utility is required for models 815, 825, 835, and 855.

Use the following procedure to install the CDA utility on a disk.

1. Ensure that the tape controlware (for 66x subsystems) and disk controlware are present and functioning properly (refer to coldstart in the NOS 2 Operator/Analyst Handbook).

2. If the disk receiving the program or data does not already have a CDA utility, ensure that the disk contains no dayfiles, permanent files, or queue files that must be preserved. The CDA utility can be installed on the same disk as CTI and HIVS.

3. Mount the operating system deadstart tape without the write enable ring and ready the unit.

4. Set the deadstart panel for warmstart from tape; for models 815 and 825, enter the values for warmstart from tape through the initial deadstart display (refer to the NOS 2 Operator/Analyst Handbook).

5. Press the deadstart switch. The initial options (A) display appears (refer to figure 2-1).

6. Select the U option. The utilities (U) display appears (refer to figure 2-2).

7. Select the C option. The following message appears.

   ```
   CTI/MSL DISK AREA UTILITY
   FOR INSTALLATION OF PROGRAMS
   AND DEFAULT PARAMETER DECK,
   AND DEFAULT PARAMETER SETTING.
   ENTER -CR- TO CONTINUE
   ```

8. Press CR. The following message appears asking you to specify the channel connected to the disk subsystem on which the CDA utility is to reside.

   ```
   ENTER DISK CHANNEL - xx
   ```

9. Enter the two-digit octal channel number of the disk subsystem, if it is different from the default value shown (xx), and press CR, or press CR to select the default channel number. The next message requests the unit number of the disk on which the CDA utility is to reside.

   ```
   ENTER DISK UNIT - xx
   ```

†The off-line MSL is provided only to the customer engineer who is assigned responsibility for a site under the contractual requirements of the Control Data maintenance services agreement. Contact your customer engineer for more information.

10. Enter the two-digit octal unit number of the disk, if it is different from the default value shown (xx), and press CR, or press CR to select the default unit number. One of the following occurs:

- If the CDA utility is installed successfully, the CDA options display appears (refer to figure 2-4) and the installation is complete.

```
SELECT DESIRED OPTION.

A - MODIFY THE DEFAULT
    PARAMETER DECK.

B - INSTALL DEFAULT PARAMETER
    DECK TO THE CTI/MSL DISK AREA.

C - INSTALL MICRO-CODE TO THE
    CTI/MSL DISK AREA.

D - INSTALL THE ENVIRONMENT
    INTERFACE TO THE CTI/MSL
    DISK AREA.
```

Figure 2-4. CDA Utility Options Display

- If the disk unit selected for the CDA utility is reserved by another controller, the following message appears.

    DISK RESERVED.

  When the reserved status of the disk unit drops, CTI automatically continues.

- If the disk unit selected for the CDA utility is a fixed module drive whose READ ONLY switch is set, the following message appears.

    READ ONLY SELECTED.

  Turn off the READ ONLY switch on the front of the drive and press CR on the console to initiate automatic retry.

- If the disk unit does not have a prior version of the CDA utility, the following message appears.

```
NO SPACE RESERVED FOR CTI/MSL
DISK AREA.  MSL/HVS OR OS FILES
MAY BE LOST IF THE OPERATION
CONTINUES.  ENTER -BKSP- TO
SELECT A DIFFERENT DISK,
or -CR- TO USE THE CURRENT DISK
```

  To install the CDA utility onto a different disk, press the backspace key and repeat steps 8, 9, and 10. To use the currently selected disk, press CR. The CDA utility options display appears; the installation is complete; and, as stated in the message, MSL/HIVS or operating system files may have been destroyed.

The following subsections describe the individual CDA utility options. Choose the A option last, because this option requires a deadstart when its installation is complete.

INSTALL DEFAULT PARAMETER DECK TO CTI/MSL DISK AREA (B OPTION)

The default parameter deck is located with the CTI module on both the deadstart tape and the HIVS tape. The parameter deck may be installed from either of these tapes. Use the following procedure.

NOTE

The messages requesting tape type, channel, equipment, and unit appear only if this option is the first option selected. CTI assumes the same configuration for subsequent options and does not request re-entry of the parameters. If any part of the configuration will be different for a subsequent option, you must deadstart the system and bring up the CDA option display again before selecting the option.

1. Select the B option from the CDA utility options display. The following message appears.

```
1. ENTER TAPE TYPE - x
   (1 = 66X, 2 = 67X)
```

2. Enter the number corresponding to the tape type of the default parameter tape and press CR, or press CR to select the default tape type x. The next message requests the number of the channel connected to the controller for the tape drive from which the default parameter tape will be loaded.

```
ENTER TAPE CHANNEL - xx
```

3. Enter the number of the channel and press CR, or press CR to select the default channel number xx. The next message requests the equipment number of the tape controller for the tape drive from which the default parameter tape will be installed.

```
ENTER TAPE EQUIPMENT - x
```

4. Enter the equipment number of the controller and press CR, or press CR to select the default equipment number x. The next message requests the unit number of the tape drive from which the default parameter tape will be installed.

```
ENTER TAPE UNIT - xx
```

5. Enter the unit number of the tape drive and press CR, or press CR to select the default unit number xx. The message INSTALLING DPB appears while the default parameter deck is being installed. When the installation is complete, the CDA utility options display appears.

INSTALL MICROCODE TO CTI/MSL DISK AREA (C OPTION)

The CDA utility allows you to install the microcode for models, 815, 825, 835, and 855 on the CDA. Use the following procedure.

NOTE

The messages requesting tape type, channel, equipment, and unit appear only if this option is the first option selected. CTI assumes the same configuration for subsequent options and does not request re-entry of the parameters. If any part of the configuration will be different for a subsequent option, you must deadstart the system and bring up the CDA option display again before selecting the option.

1. Select the C option from the CDA utility options display. The following message appears.

```
1.  ENTER TAPE TYPE - x
    (1 = 66X, 2 = 67X)
```

2. Enter the number corresponding to the tape type of the microcode tape and press CR, or press CR to select the default tape type x. The next message requests the number of the channel connected to the controller for the tape drive from which the microcode tape will be installed.

```
ENTER TAPE CHANNEL - xx
```

3. Enter the number of the channel and press CR, or press CR to select the default channel number xx. The next message requests the equipment number of the tape controller for the tape drive from which the microcode tape will be installed.

```
ENTER TAPE EQUIPMENT - x
```

4. Enter the equipment number of the controller and press CR, or press CR to select the default equipment number x. The next message requests the unit number of the tape drive from which the microcode tape will be installed.

```
ENTER TAPE UNIT - xx
```

5. Enter the unit number of the tape drive and press CR, or press CR to select the default unit number xx. The next message requests the microcode type, which corresponds to the type of computer in which the microcode will be used.

```
2.  SELECT MICRO-CODE TYPE.

        10 = 170 - 825
        11 = 170 - 815
        12 = 170 - 825E
        20 = 170 - 835
        30 = 170 - 855
```

6. Enter the number corresponding to the computer in which the microcode will be used (10 for model 825, 20 for model 835, 30 for model 855) and press CR. The message INSTALLING UmtL appears while the microcode is being installed.

Name	Description
UmtL	Microcode module name.

Parameter	Description
m	Mainframe model.

m	Model
1	815 or 825
2	835
3	855

Parameter	Description
t	Microcode type; identifies the control memory to which the segment of microcode is installed. For model 815, the type is B; for models 825 and 835, the type is A; for model 825E, the type is C; for model 855, the type is B, C, D, L, M, R, or S.

For example, the name U2AL indicates microcode for a model 835 mainframe.

When the installation is complete, the CDA utility options display appears.

INSTALL ENVIRONMENT INTERFACE TO CTI/MSL DISK AREA (D OPTION)

The CDA utility allows you to install the environment interface (EI) for models 815, 825, 835, and 855 on the CDA. Use the following procedure.

NOTE

The messages requesting tape type, channel, equipment, and unit appear only if this option is the first option selected. CTI assumes the same configuration for subsequent options and does not request re-entry of the parameters. If any part of the configuration will be different for a subsequent option, you must deadstart the system and bring up the CDA option display again before selecting the option.

1. Select the D option from the CDA utility options display. The following message appears.

   1. ENTER TAPE TYPE - x
      (1 = 66X, 2 = 67X)

2. Enter the number corresponding to the tape type of the tape containing the environment interface and press CR, or press CR to select the default tape type x. The next message requests the number of the channel connected to the controller for the tape drive from which the environment interface tape will be installed.

   ENTER TAPE CHANNEL - xx

3. Enter the number of the channel and press CR, or press CR to select the default channel number xx. The next message requests the equipment number of the tape controller for the tape drive from which the environment interface tape will be installed.

   ENTER TAPE EQUIPMENT - x

4. Enter the equipment number of the controller and press CR, or press CR to select the default equipment number x. The next message requests the unit number of the tape drive from which the environment interface tape will be installed.

   ENTER TAPE UNIT - xx

5. Enter the unit number of the tape drive and press CR, or press CR to select the default unit number xx. The message INSTALLING EI appears while the environment interface is being installed. When the installation is complete, the CDA utility options display appears.

MODIFY DEFAULT PARAMETER DECK (A OPTION)

Use the following procedure to modify the default parameters. To review parameter entries before deadstart, press the minus (-) key to display the immediately preceding entry; press the plus (+) key to display following entry. After making the desired modifications, press the deadstart switch to deadstart the system. Deadstarting is possible at any point in the procedure.

NOTE

Before deadstarting the system, ensure that you enter a carriage return after the last default parameter modification. If you do not, that parameter will not be modified.

1. Select the A option from the CDA utility options display after installing the default parameter deck. The following message appears.

```
DEFAULT PARAMETER PROCESSING

EACH ENTRY WILL BE PROCESSED
WHEN A -CR- IS ENTERED.

SPECIAL KEY INPUTS:
(+)       -DISPLAY THE NEXT
             DEFAULT BLOCK.
(-)       -DISPLAY THE LAST
             DEFAULT BLOCK.
BKSP      -DELETE LAST CHARACTER.
CR        -ENDS EACH ENTRY, WRITES
             DEFAULTS TO DISK.

ALL ENTRIES ARE IN THE FORMAT:

        XXXX

WHERE XXXX=1 TO 10 ALPHA-
NUMERIC ENTRY DEFINING THE
PARAMETER TO BE PROCESSED.

(ENTER + TO CONTINUE.)
```

2. Enter +. The following message appears.

```
DEADSTART TAPE DEFAULTS

ENTER TAPE TYPE..........xx
(1 = 66X 2 = 67X)
```

3. Enter the number corresponding to the tape type of the deadstart tape and press CR, or press CR to select the default tape type xx. The next message requests the number of the channel connected to the controller for the tape drive from which the deadstart tape will be installed.

```
ENTER CHANNEL NUMBER ....xx
```

4. Enter the number of the channel and press CR, or press CR to select the default channel number xx. The next message requests the equipment number of the tape controller for the tape drive from which the deadstart tape will be installed.

```
ENTER EQUIPMENT NUMBER ..xx
```

5. Enter the equipment number of the controller and press CR, or press CR to select the default equipment number xx. The next message requests the unit number of the tape drive from which the deadstart tape will be installed.

```
ENTER UNIT NUMBER ........xx
```

6. Enter the unit number of the tape drive and press CR, or press CR to select the default unit number xx. The following message appears.

```
ENTER + TO CONTINUE.
```

7. Enter +. The following message appears.

    EDD TAPE DUMP DEFAULTS

    ENTER TAPE TYPE..........xx
    (1 = 66X 2 = 67X)

8. Enter the number corresponding to the tape type of the EDD tape and press CR, or press CR to select the default tape type xx. The next message requests the number of the channel connected to the controller for the tape drive to which the EDD dump will be written.

    ENTER CHANNEL NUMBER xx

9. Enter the number of the channel and press CR, or press CR to select the default channel number xx. The next message requests the equipment number of the tape controller for the tape drive to which the EDD dump will be written.

    ENTER EQUIPMENT NUMBER ..xx

10. Enter the equipment number of the controller and press CR, or press CR to select the default equipment number xx. The next message requests the unit number of the tape drive to which the EDD dump will be written.

    ENTER UNIT NUMBER xx

11. Enter the unit number of the tape drive and press CR, or press CR to select the default unit number xx. The following message appears.

    ENTER + OR - TO CONTINUE.

12. Enter +. The following message appears.

    SYSTEM DISK DEFAULTS

    ENTER CHANNEL NUMBER xx

13. Enter the number of the channel connected to the system disk and press CR, or press CR to select the default channel number xx. The next message requests the unit number of the system disk.

    ENTER UNIT NUMBER xx

14. Enter the unit number of the disk and press CR, or press CR to select the default unit number xx. The following message appears.

    ENTER + OR - TO CONTINUE.

15. Enter +. The following message appears.

    ALTERNATE SYSTEM
      DISK DEFAULTS

    ENTER CHANNEL NUMBER xx

16. Enter the number of the channel connected to the alternative system disk and press CR, or press CR to select the default channel number xx. The next message requests the unit number of the alternative system disk.

    ENTER UNIT NUMBER xx

17. Enter the unit number of the disk and press CR, or press CR to select the default unit number xx. The following message appears.

    ENTER + OR - TO CONTINUE.

18. Enter +. The following message appears.

    MSL-HIVS DISK DEFAULTS

    ENTER CHANNEL NUMBER xx

19. Enter the number of the channel connected to the disk subsystem on which the HIVS module is to reside and press CR, or press CR to select the default channel number xx. The next message requests the unit number of the disk on which the HIVS module is to reside.

    ENTER UNIT NUMBER xx

20. Enter the unit number of the disk and press CR, or press CR to select the default unit number xx. The following message appears.

    ENTER + OR - TO CONTINUE.

21. Enter +. The following message appears.

    LINE PRINTER DUMP DEFAULTS

    ENTER CHANNEL NUMBER xx

22. Enter the number of the channel connected to the line printer to be used for line printer dump and press CR, or press CR to select the default channel number xx. The next message requests the equipment number of the line printer.

    ENTER EQUIPMENT NUMBER ..xx

23. Enter the equipment number of the line printer and press CR, or press CR to select the default equipment number xx. The next message requests the train type of the line printer.

    ENTER TRAIN TYPE........xx
    (1 = 595-1, 2 = 596-2)
    (3 = 596-3, 4 = 596-4)
    (5 = 595-5, 6 = 595-6)

24. Enter the number corresponding to the train type† of the line printer and press CR, or press CR to select the default train type xx. The next message requests the format control mode for the line printer.

    FORMAT CONTROL MODE xx
    1 = PROGRAMMABLE   2 = FORMAT TAPE

25. Enter the number corresponding to the format control mode for the line printer and press CR, or press CR to select the default mode xx. The following message appears.

    ENTER + OR - TO CONTINUE.

†NOS 2 does not support train types 596-2, 596-3, nor 596-4.

26. Enter +. The following message appears.

```
PARAMETER PROCESSING COMPLETE
ENTER (-), OR DEADSTART
```

27. Press the deadstart switch to deadstart the operating system.

STEP 6 INSTALL HIVS MODULE

The HIVS module performs hardware confidence tests; it appears as an option in the CTI displays. For model 815, 825, 835, or 855, HIVS must be installed on a disk in order to deadstart NOS. Installing HIVS on a disk allows the running of the hardware verification sequence in subsequent deadstarts. Do not install HIVS if MSL resides on the disk, because MSL provides the hardware verification sequence.

Use the following procedure to install HIVS. If the disk that receives the HIVS module is already a system disk, installing HIVS leaves the operating system intact.

1. Ensure that the tape controlware (for 66x subsystems) and disk controlware are present and functioning properly (refer to coldstart in the NOS 2 Operator/Analyst Handbook).

2. If the disk receiving the HIVS module does not already have the desired CTI module, or if the contents of the disk are unknown, install the desired CTI module on this disk. (Refer to Install CTI Module in this section.)

3. Mount the HIVS installation tape without the write enable ring, and ready the unit.

4. Set the deadstart panel for warmstart from tape; for models 815 and 825, enter the values for warmstart from tape through the deadstart display (refer to the NOS 2 Operator/Analyst Handbook).

5. Press the deadstart switch. The initial options (A) display appears (refer to figure 2-1).

6. Select the U option. The utilities (U) display appears (refer to figure 2-2).

7. Select the T option. The display that appears requests the number of the channel connected to the 844 or 885 disk subsystem upon which HIVS will reside.

```
ENTER PARAMETERS
DISK CH xx
```

8. Enter the channel number of the 844 or 885 disk subsystem, if it is different from the default value xx, and press CR. The next display requests the unit number of the disk.

```
DISK UN xx
```

9. Enter the number of the disk unit that is to receive HIVS, if it is different from the default value xx, and press CR. The next display requests the type of tape unit.

```
TAPE TYPE xx

0=60X,1=65X,2=66X,3=67X
```

10. Select the type of tape unit,† if it is different from the default value xx, and press CR. The next display requests the channel number of the tape controller or 66x/67x subsystem on which the HIVS installation tape is mounted.

    TAPE CH xx

11. Enter the channel number of the tape controller or 66x/67x tape subsystem, if it is different from the default value xx, and press CR. The next display requests the equipment number of the tape controller or 66x/67x tape subsystem.

    TAPE EQ xx

12. Enter the equipment number of the 66x/67x tape subsystem, if it is different from the default value xx, and press CR (the equipment number for tape type 66x is 00 and for tape type 67x is from 00 through 07). The next display requests the unit number of the tape drive.

    TAPE UN xx

13. Enter the unit number of the tape drive on which you mounted the HIVS tape, if it is different from the default value xx, and press CR.

    While HIVS is being loaded to disk, first INITIALIZING and then LOADING is displayed on the left side of the display screen.

    If you want to stop the system from copying programs to the disk, press the S key; if you want to resume copying programs to the disk, press the space bar.

    Upon completion of installing HIVS on the disk from tape, the following message appears.

    END

14. Press the deadstart switch. The HIVS installation is complete.

STEP 7 CREATE VALIDATION AND PROJECT PROFILE FILES

Create the validation and project profile files, as necessary. Refer to the descriptions of the GENVAL procedure file, the PROFILE command, and the MODVAL command in the NOS 2 System Maintenance Reference Manual.

---

†NOS 2 does not support 60x and 65x tape units.

STEP 8 SET UP INSTALLATION FILES

Generate the permanent files required for the installation process by running the following jobs. Steps 2 and 3 are optional.

1. Run a job similar to the following to create file DECKOPL, procedure file INSTALL, and other files and procedures used in the installation process.

Job	Comments
job command.	
USER,username,password,familyname.	
CHARGE,*.	
LABEL,TAPE,D=PE,VSN=RELO.	Change the tape density option if RELO is not D=PE (PE is nine-track and 1600 cpi).
BEGIN,,TAPE.	
--eoi--	

2. Run a job similar to the following to obtain the listings of all installation procedures on DECKOPL. Refer to table 2-1 for a list of the product procedure names.

```
job command.
USER,username,password,familyname.
CHARGE,*.
BEGIN,DECKLIS,INSTALL.
--eoi--
```

INCLUDE THE PARAMETER /PLUSPF TO LIST COMPILE TIME DIRECTIVES, OTHERWISE THEY ARE EXPANDED

3. Run a job similar to the following to modify any of the installation procedures on file DECKOPL.†

job command.	
USER,username,password,familyname.	
CHARGE,*.	
COPYBR,INPUT,DECKMOD.	
REWIND,DECKMOD.	
BEGIN,SETUP,INSTALL,NEWPL,MOD=DECKMOD.	If DECKMOD is not local, it must be a permanent file.
--eor--	
(Modifications on the input record.)	
--eoi--	

4. ~~Run a job similar to the following to print the PSR summaries from RELO. This job automatically requests RELO. Specify the D=density parameter if RELO is not the default density (D=PE).~~

```
job command.
USER,username,password,familyname.
CHARGE,*.
BEGIN,PSRLIST,INSTALL,REPORT=p,D=density.
--eoi--
```

†Refer to UCOMMOD in example 6 under SETUP Procedure in this section for changes to common deck parameters.

The following lists are printed, depending on the value of p. If the REPORT parameter is omitted, REPORT=0 is assumed.

p	Type of List
0	System PSR list.
1	PSRs sorted by product and then by PSR number.
2	PSRs sorted by product and then by routine.
3	PSRs sorted by product and then by site.

63-CHARACTER SET INSTALLATION

The system can be installed in 63-character set format on either a dedicated system or a system running other jobs (production environment).

To install the 63-character set format on a dedicated system, perform the following steps.

1. Run a job that includes the following command:

   ```
   BEGIN,SETUP,INSTALL,DF63,INSTALL.
   ```

2. Continue the installation process until procedure COMBINE is complete. The composite OPL (RELSOPL) will be in 63-character set format.

3. Deadstart again, adding the following IPRDECK entry.

   ```
   CSM=63.
   ```

4. Execute the following command to convert DECKOPL to 63-character set format and create a new file INSTALL.

   ```
   BEGIN,SETUP,INSTALL,CV63,DF63,INSTALL,NEWPL.
   ```

5. Continue the installation procedure and include the following on file USER for the call to TEXT.

   ```
   *IDENT 63CSET
   *I,IPARAMS.15
    IP.C63  EQU   IP.C64.1
    IP.CSET EQU   IP.C63
   */      END OF MODSET.
   ```

6. Complete the installation procedures as documented.

To install the 63-character set format in a production environment (it is assumed that the existing system is already in 63-character set format), perform the following steps.

1. Run a job that includes the following command.

   ```
   BEGIN,SETUP,INSTALL,DF63,CV63,INSTALL.
   ```

2. Continue the installation procedure and include the following on file USER for the call to TEXT.

```
*IDENT 63CSET
*I,IPARAMS.15
IP.C63  EQU  IP.C64.1
IP.CSET EQU  IP.C63
*/      END OF MODSET.
```

3. Complete the installation procedures as documented.

NOTE

Unpredictable and possibly serious problems occur if the operating system is operating in one character set and the common product set is operating in another. Therefore, ensure that all installed products and the operating system are in the same character set mode.

Table 2-1. Product Installation Information (Sheet 1 of 3)

Product	Procedure Name		Product† Format
	Installation	Verification	
Base Operating System			
BINEDIT	BINEDIT		Update
CEDIAG 1	CEDIAG		Update
Common Memory Manager 1	UPDATE		Update
COMPASS 3	COMPASS		Update
CYBER Common Utilities	UPDATE		Update
CYBER Control Language 1	CCL		Update
CYBER Loader 1	LOADER		Update
CYBER Record Manager			
Basic Access Methods 1	BAM	VBAM	Update
Advanced Access Methods 2	AAM2	VAAM2	Update
FORM 1	FORM	VFORM	Update
Network Operating System 2	COMBINE, NOS		Modify
Product Texts	TEXT		Update
Product Texts I/O	TEXTIO		Update
Update 1	UPDATE		Update
8-Bit Subroutines 1	BIT8	VBIT8	Update
667/669 Controlware	MTS		-
844 Controlware	BCF		-
	BCS		-
885-11/12 Controlware	FMD		-
885-42 Controlware	PHD		-
819 PPU Driver	HCD		Update

† - indicates not applicable.

Table 2-1. Product Installation Information (Sheet 2 of 3)

Product	Procedure Name: Installation	Procedure Name: Verification	Product Format
Maintenance Package			
CYBRLOG	SYSJOB		-
Maintenance Tools	COMBINE, TOOLS		Modify
SYMPL 1	SYMPL	VSYMPL	Update
881/883 Pack Formatting	FORMAT		Update
Optional Products			
ALGOL-60 5	ALGOL5	VALGOL5	Update
APL 2	APL2	VAPL2	Modify
BASIC 3	BASIC3	VBASIC3	Update
COBOL 5	COBOL5	VCOBOL5	Update
Common Code Generator	CCG		Update
Communications Control Program 3			
Initialization and MUX Firmware	CCPPH1		Update
Binary Library Build	CCPBLB		Update
Variant Load Module	CCPVAR		Update
Patch Load Module	CCPEDIT		Update
Load File Generation	CCPLOAD		Update
File Cleanup	CCPPURG		Update
Conversion Aids System 3	LCS3 FCS3	VLCS3 VFCS3	Update
CYBER Cross System 1	CROSS	VCROSS	Update
CYBER Database Control System 2	CDCS2	VCDCS2A VCDCS2B	Update
CYBER Interactive Debug 1	CID	VCID	Update
Data Catalogue 2	DCAT2		Update
Data Description Language 3	DDL3	VDDL3	Update
Database Utilities 1	DBU	VDBU	Update
FORTRAN Common Library 4	FCL1 FCL2		Update
FORTRAN Common Library 5	FCL5		Update
FORTRAN Data Base Facility 1	FDBF	VFDBF	Update
FORTRAN Extended 4	FTN	VFTN	Update
FORTRAN Extended 4 with Interactive Option	FTNTS	VFTNTS	Update
FORTRAN 4 Postmortem Dump Utility	PMD4†		Update
FORTRAN 4 to 5 Conversion Aid	F45	VF45	Update
FORTRAN 5	FTN5	VFTN5	Update
FORTRAN 5 Postmortem Dump Utility	PMD5†		Update
High Speed I/O Option	HSIO		Modify
Information Management Facility 1	IMF1	VIMF1	Update

†Both PMD4 and PMD5 produce functionally equivalent postprocessors. Install only one of them.

Table 2-1. Product Installation Information (Sheet 3 of 3)

Product	Procedure Name		Product Format
	Installation	Verification	
Optional Products (Contd)			
Interactive Facility 1	COMBINE, IAF		Modify
Mass Storage Subsystem	COMBINE, MSS		Modify
Message Control System 1	MCS	VMCS1A VMCS1B	Update
Multimainframe Module 1	COMBINE, MMF		Modify
Network Access Method 1	NAM5		Update
PASCAL 170	PASCAL	VPASCAL	Update
PL/I 1	PLI	VPLI	Update
Query Update 3	QU3	VQU3	Update
Remote Batch Facility 1	RBF5		Update
Remote Host Facility	RHF RHP		
Sort/Merge 4	SORT	VSORT	Update
Sort/Merge 5	SORT5	VSORT5	Update
Tracer 1	COMBINE, TRACER		Modify
Transaction Facility 1	COMBINE, TAF		Modify
XEDIT 3	COMBINE, XEDIT		Modify
380-xxx Controlware	NAD MIN IBM		

STEP 9 CREATE USER FILES

File USER allows you to modify the products during installation. The modifications on file USER must be in the same format (Modify or Update) as the product. Refer to table 2-1 to determine a product's format. Modifications can include:

- Adding PSR corrective code.
- Adding code from file MISCPL (refer to the description of RELO in appendix C).
- Changing the installation parameter settings (refer to the Product Modifications part).
- Adding site code.

During installation procedure execution, local modifications to the product must be contained in the local file USER. The preferred method of doing this (the only method for interactive calls) is to store the local modifications on a permanent file (for example, USERAAM) and to pass the file name on the call to the installation procedure, as in the following command.

```
BEGIN,AAM2,INSTALL,USERF=USERAAM.
```

Two other options for batch jobs that call the installation procedure are:

- ```
  GET,USER=USERAAM.
  BEGIN,AAM2,INSTALL.
  ```

- ```
  GET,USERAAM.
  BEGIN,AAM2,INSTALL,USERF=USERAAM.
  ```

All code that may correct a specific user site problem but which has not been fully tested is contained on the file MISCPL, which was created during the setup of installation files (step 8). The modifications are properly formatted for the intended program library (Update or Modify). To list the modification set (modset) headers and the decks containing /CALLs to common decks containing modsets, use the following commands.

```
BEGIN,MISCGET,INSTALL,HISTORY.
ROUTE,USER,DC=PR.
```

Use the listing to select the decks you want to extract from MISCPL.

Code can be extracted from MISCPL in one of the following ways.

- All modsets for a product.

  ```
  BEGIN,MISCGET,INSTALL,PRD=name.
  ```

  name is the name of the deck containing a /CALL for each modset available for the product.

- One modset.

  ```
  BEGIN,MISCGET,INSTALL,MOD=modname.
  ```

  modname is the name of the required modset.

- Selected modsets.

  ```
  NOTE,lfn,NR.+,modname+,modname+,...,modname.

  BEGIN,MISCGET,INSTALL,MISCIN=lfn.
  ```

These procedures append the modsets to the local file USER. The completed file USER must be packed with the PACK command if it is built for an Update formatted product. Refer to the NOS 2 Reference Set, Volume 3, for a description of the PACK command.

DOES NOT HAVE TO BE PACKED AFTER MISCGET HAS ADDED MODSETS TO IT

STEP 10 INSTALL PRODUCTS

Listed in table 2-1 are the products, their installation and verification procedure names, and their format (Modify or Update). Because dependencies exist between installation jobs, they are separated into three groups for installation: group 1 products, group 2 products, and group 3 products (refer to tables 2-2, 2-3, and 2-4). Descriptions of build procedures and the installation of groups 1, 2, and 3 follow.

For a partial installation of the products, check figure 2-5 to determine the procedures on which the products depend.

Direct access file PRODUCT contains the binaries produced by the installation procedures. If many products are installed and it becomes necessary to make more disk space available, generate a new deadstart tape using GENSYS (refer to Create Deadstart Tape in this section). GENSYS incorporates the binaries in PRODUCT into the new deadstart tape. You can reinitialize PRODUCT and GLOBLIB by attaching the latest deadstart tape as local file DST and executing the following command.

SEE ERRATA

```
BEGIN,SEED,INSTALL,DST=lfn,REBUILD
```

SETTING UP THE INSTALLATION ENVIRONMENT

Procedure TAPE (step 8) stores from DECKOPL, all of the installation and verification procedures on a direct access file named INSTALL. This file has an OPLD type directory for random access to procedures.

DECKOPL contains installation procedures for all products, verification procedures for most products, common utility procedures used by installation procedures, utility procedures available to you, and common decks containing formal parameters for the installation procedure PROC declarations.

The common decks are useful for defining global defaults for the installation procedures. For example, default tape density can be changed or standard file names can be changed to allow separate installations under the same user index. Refer to the examples under SETUP Procedure for further information.

Setup Procedure

The support utility procedure SETUP resides on INSTALL and is used to:

- Initialize files.
- Create INSTALL from DECKOPL with optional modifications against DECKOPL.
- Rename file INSTALL.
- Create a 63-character set version of procedures on INSTALL.
- Convert DECKOPL to a 63-character set Modify format.
- Replace DECKOPL with an updated DECKOPL.

The SETUP procedure call has the following format.

```
BEGIN,SETUP,INSTALL,RESET,NEWPL,MOD=modfil,DF63,CV63,INSTALL=pfile.
```

All parameters are optional; if none are specified, SETUP will run but performs no useful work.

Parameter	Description
RESET	Initializes direct access files DAYFILS and JOBSTAT. If RESET is omitted, the files are not initialized.
NEWPL	Replaces DECKOPL with modified DECKOPL. If NEWPL is omitted, DECKOPL is not replaced.
MOD=modfil	Applies modsets from file modfil to DECKOPL; modfil can be a local file or a direct or indirect access permanent file.
DF63	Selects 63-character set version of installation procedures for inclusion on file INSTALL.
CV63	Converts DECKOPL to 63-character set format.
INSTALL=pfile	Creates or replaces procedure file pfile. The default for pfile is INSTALL. If you omit the INSTALL keyword, procedure file INSTALL is not created or replaced.

Example 1. The following command initializes files before building a new system.

```
BEGIN,SETUP,INSTALL,RESET.
```

Example 2. The following command builds file INSTALL with modifications on file MODFIL against DECKOPL. This format can be used to change default parameters in the common decks. It does not replace DECKOPL with the modified DECKOPL.

```
BEGIN,SETUP,INSTALL,MOD=MODFIL,INSTALL.
```

Example 3. The following command applies modsets on permanent file XXX to DECKOPL and replaces DECKOPL with the modified DECKOPL. It does not replace INSTALL.

```
BEGIN,SETUP,INSTALL,NEWPL,MOD=XXX.
```

Example 4. The following command creates the file INSTALL.

```
BEGIN,SETUP,INSTALL,INSTALL.
```

Example 5. The following command replaces the file INSTALL with 63-character set versions of installation procedures that are sensitive to 63-character set installation.

```
BEGIN,SETUP,INSTALL,DF63,INSTALL.
```

Example 6. The following command builds a modset on file COMMOD for changing global defaults in common decks.

```
BEGIN,UCOMMOD,INSTALL.
```

Edit the modset file, changing the defaults for all parameters that require change. Then execute the following command to produce a new file INSTALL with the changed parameters.

```
BEGIN,SETUP,INSTALL,MOD=COMMOD,INSTALL.
```

RUNNING INSTALLATION JOBS

You can call all installation procedures interactively or by batch jobs. If you call them interactively, the procedure submits a batch job for execution. (If you require file USER for an interactively called procedure, you must include the parameter USERF=pfn on the BEGIN command.) Include the USER and CHARGE commands on file USERCHG when you call the procedures interactively.

The installation procedures use the global library set to allow installation in a production environment under NOS 2.

Each installation procedure makes the global library file (GLOBLIB) a local file by executing the procedure ATTGLOB. Those DECKOPL procedures that produce binaries required by subsequent installation procedures signal the procedure END that certain binaries should be added to GLOBLIB. END creates a temporary permanent file from the current GLOBLIB and the new binaries, purges GLOBLIB, changes the name of the temporary file to GLOBLIB, and returns GLOBLIB. This method requires some overhead in the form of disk space for those installation jobs still executing with the old copy of GLOBLIB still attached.

INSTALLATION PROCEDURE PARAMETERS

Parameters for the installation procedures are of two types:

- Parameters that are unique to the product.
- Parameters that are common to two or more products.

The parameter format is keyword or keyword=value. Where only the keyword is listed, the presence of the keyword in the procedure call selects the option. Where the keyword=value format is listed, you must supply a value in the call to change the parameter. ~~These parameters do not apply to Cross and CCP installation procedures.~~

The format of an installation procedure call is:

```
BEGIN,procname,INSTALL,unique parameters,common parameters.
```

Unique Parameters

The unique parameters are listed in tables 2-2, 2-3, and 2-4. Refer to the procedure names in section 5 for the meanings of these parameters.

Common Parameters

Common parameters are contained in common decks on DECKOPL. Following is a description of the common parameters of interest. Execute procedure UCOMMOD and print file COMMOD to obtain a listing of all common parameters.

Parameter	Description
PC=pc	Controls the inclusion of code to update the released program libraries from the previous release to the PSR summary level of the current release. If the product program library was included with the current installation materials, PC=NO should be used. The default is PC=NO.

pc	Significance
YES	Updates program library to current PSR summary level.
NO	Does not update the release program library.

Parameter	Description
CC=yy	Controls the inclusion of corrective code in the compilations when building the product. This code is not added to the output program library. The default is CC=YES.

yy	Significance
YES	Includes corrective code.
NO	Does not include corrective code.

Parameter	Description
BINARY	Includes the binary file from the product's REL tape on file PRODUCT. When BINARY is specified, no modifications can be made to that product. File USER is not used.

NOTE

It may be incorrect to specify BINARY for a product if you have modified any system texts, such as NOSTEXT or IPTEXT. Some products use symbolic constants in communicating with the operating system or other products. The values of these symbolic constants are usually defined in system texts and, if the values of the symbolic constants are modified, all products that reference them must be reassembled. If problems arise, try reinstalling the product without the BINARY parameter.

Parameter	Description
USERF=pfn	Specifies the file name of the modification file. If site modifications are on a file named USER that is local to the installation batch job (not called interactively), this parameter should not be used. If the modifications are on the permanent file pfn and that file is not local to the batch job, the procedure makes it a local file. If pfn is a local file, it is renamed as file USER.

Parameter	Description
	If you add site modifications through file USER, the code is applied last, is added to the generated binaries, and is not installed on the program library. If this program library is to be used as input to another product, all code is not present and problems may result. Creation of a new program library with your site modifications outside this installation job may be required.
~~DISKINS~~	~~Specifies installation from mass storage. Routines TAPEIN, TAPEAUX, and TAPEOUT will attach the proper files from mass storage.~~
D=density	Specifies the tape density option. If this parameter is omitted, D=PE is the default.

density	Significance
HI	556 cpi (seven-track)
HY	800 cpi (seven-track)
HD	800 cpi (nine-track)
PE	1600 cpi (nine-track)
GE	6250 cpi (nine-track)

LIST=filename	Specifies the assembly or compilation listing destination. If this parameter is omitted, no listing is produced. If filename is OUTPUT, the listing will be printed with the installation listing. If you specify any other file name, disposition of the listing is determined by the TOLIST parameter.
TOBLD=b	Specifies the build listing disposition; determines whether the job output goes to the terminal output queue. The default is TOBLD=NO.

b	Significance
YES	Job output is placed in the terminal output queue with a user job name (UJN) equal to the installation procedure name followed by the letter B (for example, AAM2B). If the procedure name is seven characters, the last character is replaced by B.
NO	Job output is printed at the central site. This is also the disposition if the job fails.

Parameter	Description
TOLIST=r	Specifies the list file routing; determines whether the list file goes to the terminal output queue. The default is TOLIST=NO.

r	Significance
YES	The file named in the LIST=filename parameter is routed to the terminal output queue with the UJN equal to the installation procedure name followed by the letter L (for example, AAM2L). If the procedure name is seven characters, the last character is replaced by L.

r	Significance
NO	The list file, if defined, is local to the job and is discarded when the job terminates.

If the job fails, the list file is returned.

NOTE

Exercise care when using the terminal output queue for assembly and build listings. These queued files count toward the total number of jobs validated for your user name.

If your site needs different options for the LIST, TOBLD, and TOLIST functions, recode the procedures named JOBPASS and JOBFAIL in DECKOPL. These procedures do not apply to Cross and CCP installations.

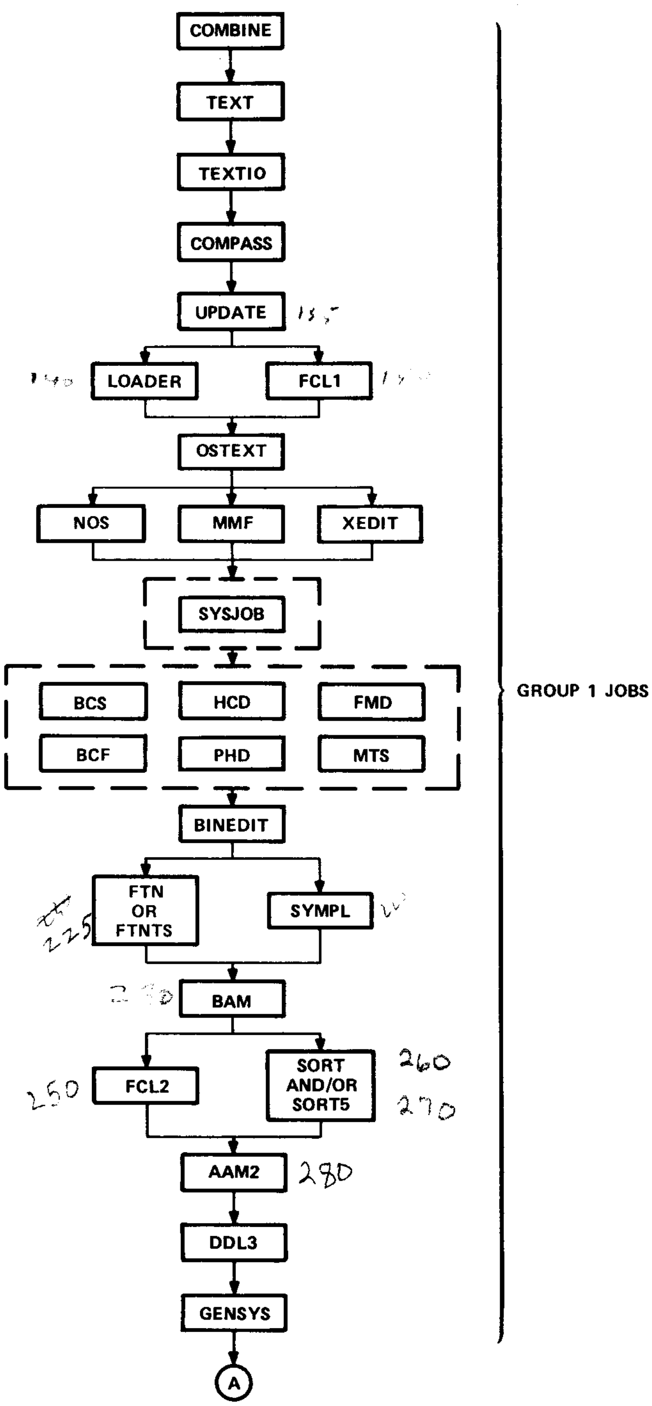

Figure 2-5. Installation Job Dependencies (Sheet 1 of 2)

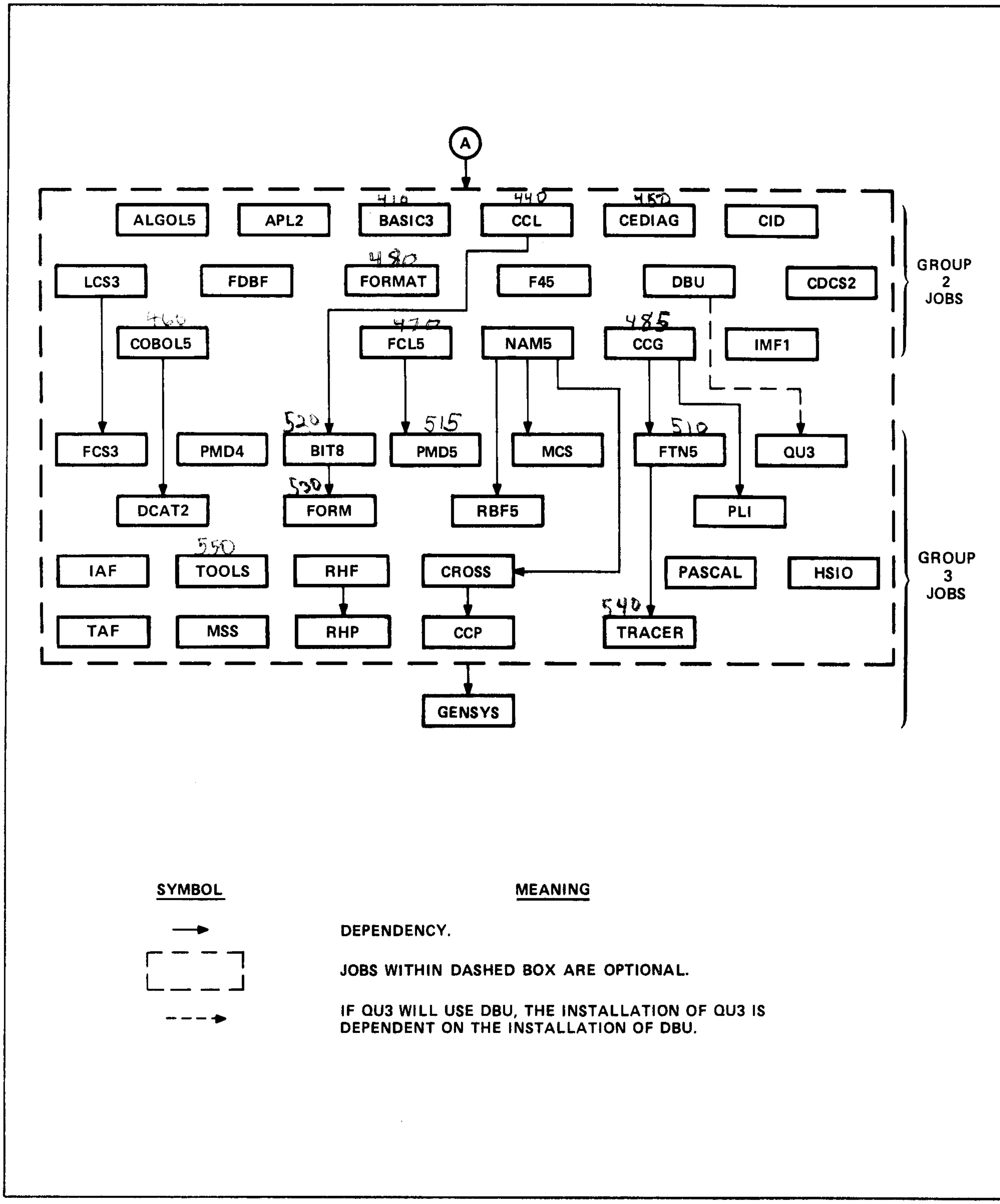

Figure 2-5. Installation Job Dependencies (Sheet 2 of 2)

To install the products, run a job similar to the one shown in figure 2-6.

Job	Comments
job command.	
USER,username,password,familyname.	
CHARGE,*.	
GET,pfn.	The GET command is optional. If it is used, the BEGIN command must use the USERF=pfn parameter. If the GET command is omitted, the USERF=pfn parameter causes the procedure pname to get or attach file pfn as a local file named USER.
BEGIN,pname,INSTALL,USERF=pfn.	
--eoi--	

Figure 2-6. Format of an Installation Job

INSTALL GROUP 1

The group 1 procedures are to be run in the order in which they are listed in table 2-2; no procedure should be started until the preceding procedure has finished.

NOTE

You may need to create additional deadstart tapes to reduce the size of the file PRODUCT if you are building on a system with limited resources.

The product procedure names for group 1, their suggested parameter settings, and their required tapes are listed in dependency table 2-2. In the following list, a short description is given for each column in the table.

Column	Description
Procedure Name	Name of the installation procedure for each product to be installed. Use this procedure name in the call to the procedure.
Keywords	All unique keywords and codes for selected common keywords that may be included in calls to installation procedures. Footnotes describe the meaning of the codes.
Input Tape VSN	Volume serial number of the release tape of the product to be installed. This tape must be available when installing the product, unless you select DISKINS. If you do, the tape contents must be on mass storage.

Column	Description
Other Tapes Required	In some cases, the installation procedure requests tapes other than the tape of the product being installed. For each applicable product, these tapes are listed in the order in which they are requested. If you select DISKINS, the procedures obtain the tape contents from mass storage.
Last Job That Used The Input Tape	Certain tapes contain more than one product or are used by more than one installation procedure. The procedure name specified in this column previously used the REL tape of the product being installed; thus, the output tape of that procedure, instead of the REL tape, must be used as the input tape for this procedure. This procedure is automatic if DISKINS is selected.
Comments	This column contains notes or additional restrictions.

COMBINE Procedure

The COMBINE procedure generates the composite old program library, RELSOPL. For the installation of the COMBINE procedure, create file USER to include:

- Suggested code (optional) from MISCPL.
- Modifications of the NOS installation parameters (refer to section 6).
- Modifications to the operating system products (refer to the installation procedures in section 5 for a description of the keywords).
- Site code.

The command BEGIN,COMBINE,INSTALL must include a keyword for each product to be included in the generated RELSOPL. Any or all of the following keywords can be used (the keywords appear in ascending REL tape number in order to facilitate tape handling.

Product	Keywords	Corresponding Release Tape VSN
Extended Interactive Text Editor	XEDIT	REL1D
Multimainframe Module	MMF	REL1F
Tracer and Probe	TRACER	REL1G
High Speed I/O Option	HSIO	REL1J
Maintenance Tools	TOOLS	REL2A
Transaction Facility	TAF	REL12C
Interactive Facility	IAF	REL12E
Mass Storage Subsystem	MSS	REL14C

There is no keyword NOS: the operating system is assumed to be included.

Run a job similar to the following for the COMBINE procedure.

```
job command.
USER,username,password,familyname.
CHARGE,*.
GET,USER.
BEGIN,COMBINE,INSTALL,keywords.
--eoi--
```

Subsequent calls to the installation procedures named as keywords, (for example, BEGIN,TAF,INSTALL,...) result in assembly or compilation of the products.

Table 2-2. Group 1 Products (Sheet 1 of 3)

Procedure Name	Keywords†		Input Tape VSN	Other Tapes Required	Last Job That Used The Input Tape	Comments
	Common	Unique				
COMBINE	CC††	HSIO, IAF, TAF, MSS, TOOLS, TRACER, MMF, XEDIT,	REL1A	†††		Output is RELSOPL.
TEXT	B, PC		REL3A			
TEXTIO	B, PC		REL3A		TEXT	
COMPASS	B, PC		REL3A		TEXTIO	
UPDATE	B, PC		REL3A		COMPASS	
LOADER	B, PC		REL3A		UPDATE	
FCL1	PC		REL4C	REL3A		No output tape. Use FCL2 to install binaries.
OSTEXT	U		RELSOPL			No output tape.
NOS	U	NOIAF	RELSOPL			No output tape.
MMF	B, U		REL1F RELSOPL			No output tape. REL1F required for binary installation, RELSOPL required for build.
XEDIT	B, U		REL1D RELSOPL			No output tape. REL1D required for binary installations, RELSOPL for build.

†In the common keywords column, B indicates that the BINARY keyword is allowed; PC indicates that permanent code (PC keyword), corrective code (CC keyword), and site code (USERF keyword) are all allowed for the named procedure; and U indicates that only code from file USER is allowed. Refer to Installation Procedure Parameters in this section for common keywords. Refer to the corresponding procedure name in section 5 for a description of unique keywords in this table.
††For the COMBINE procedure, only corrective code from MDYMODS and user code is allowed.
†††Depends on products being installed.

Table 2-2. Group 1 Products (Sheet 2 of 3)

Procedure Name	Keywords† Common	Keywords† Unique	Input Tape VSN	Other Tapes Required	Last Job That Used The Input Tape	Comments
SYSJOB††						No output tape.
BCS						No output tape. Installs new controlware (refer to section 4).
BCF						No output tape. Installs new controlware (refer to section 4).
FMD						No output tape. Installs new controlware (refer to section 4).
PHD						No output tape. Installs new controlware (refer to section 4).
MTS			REL2A	RELSOPL	COMBINE	No output tape. Installs new controlware (refer to section 4). Punches the ABC controlware loader.
HCD	B, PC		REL3A		LOADER	Installs 819 PPU driver on model 176 (refer to section 4).
170						No output tape. Installs NAD controlware (refer to section 4).
200						No output tape. Installs NAD controlware (refer to section 4).
IBM						No output tape. Installs NAD controlware (refer to section 4).

†In the common keywords column, B indicates that the BINARY keyword is allowed and PC indicates that permanent code (PC keyword), corrective code (CC keyword), and site code (USERF keyword) are all allowed for the named procedure. Refer to Installation Procedure Parameters in this section for common keywords. Refer to the corresponding procedure name in section 5 for a description of unique keywords in this table.

††Install SYSJOB (from DECKOPL) to enable CYBRLOG. If CYBRLOG is not required for subsequent deadstarts, do not install SYSJOB. Permanent file SYSJOB, created by this procedure, must reside under the system user index (377777_8) to initiate CYBRLOG. If SYSJOB is installed from a system (SY) service class, the procedure automatically saves permanent file SYSJOB under the system user index. As a result, all subsequent deadstarts initiate CYBRLOG.

Table 2-2. Group 1 Products (Sheet 3 of 3)

Procedure Name	Keywords† Common	Keywords† Unique	Input Tape VSN	Other Tapes Required	Last Job That Used The Input Tape	Comments
MIN						No output tape. Installs NAD controlware (refer to section 4).
BINEDIT	B, PC		REL3A	REL3A	HCD	Output REL3A from HCD is referenced twice for input.
SYMPL	B, PC		REL2A		MTS	
FTN	B, PC		REL4A	REL3A		Omit if FTNTS is installed.
FTNTS	B, PC		REL4B	REL3A		Omit if FTN is installed.
BAM	B, PC		REL3A	REL3A	BINEDIT	Output REL3A from BINEDIT is referenced twice for input.
FCL2	B, PC		REL4C	REL3A		
SORT	B, PC		REL6A			
SORT5	B, PC		REL6B			
AAM2	B, PC	SRT5	REL3A		BAM	
DDL3	B, PC		REL11H	REL3A		
GENSYS						If you are building on a dedicated system with limited disk space, you may want to build a new deadstart tape at the end of the group 1 installation (refer to Step 3 - Create Deadstart Tape in this section).

†In the common keywords column, B indicates that the BINARY keyword is allowed and PC indicates that permanent code (PC keyword), corrective code (CC keyword), and site code (USERF keyword) are all allowed for the named procedure. Refer to Installation Procedure Parameters in this section for common keywords. Refer to the corresponding procedure name in section 5 for a description of unique keywords in this table.

INSTALL GROUP 2

Do not run the group 2 procedures (table 2-3) until all procedures in group 1 are completed (refer to Install Group 1 for descriptions of table columns). Procedures in group 2 are not order-dependent, and they may run simultaneously.

The installation jobs are the same for all the group 2 products (refer to figure 2-6), except for the NAM2 procedure.

Table 2-3. Group 2 Products (Sheet 1 of 2)

Procedure Name	Keywords†		Input Tape VSN	Other Tapes Required	Last Job That Used The Input Tape	Comments
	Common	Unique				
ALGOL5	B, PC		REL7B			
APL2 APLUSR0 APLUSR1	B, PC	TERMTYP=typ IUN (SEE ERRATA)	REL8B	RELSOPL		APL 2 consists of three installation procedures: APL2, APLUSR0, and APLUSR1 (refer to APL2 in section 5).
BASIC3	B, PC		REL8A	RELSOPL REL3A		
CCG	PC		REL14B			CCG does not install binaries.
CCL	B, PC		REL3A	REL3A	AAM2	Output REL3A from AAM2 is referenced twice for input.
CDCS2	B, PC	DEBUG, SRT5	REL11G	REL3A REL11H RELSOPL		
CEDIAG	B, PC		REL2B	RELSOPL		
CID	B, PC		REL3F	RELSOPL		
COBOL5††	B, PC	TAF	REL5C			REQUIRES SORT & CDCS2
COBOL5Q††	B, PC	TAF	REL5C			
DBU	B, PC		REL11D			

†In the common keywords column, B indicates that the BINARY keyword is allowed and PC indicates that permanent code (PC keyword), corrective code (CC keyword), and site code (USERF keyword) are all allowed for the named procedure. Refer to Installation Procedure Parameters in this section for common keywords. Refer to the corresponding procedure name in section 5 for a description of unique keywords in this table.

††Closely monitor the installation of COBOL because problems with mass storage may arise. For example, COBOL5 creates a large compile file which could exceed track limits if more than one product is being installed.

Table 2-3. Group 2 Products (Sheet 2 of 2)

Procedure Name	Keywords† Common	Keywords† Unique	Input Tape VSN	Other Tapes Required	Last Job That Used The Input Tape	Comments
FCL5	B, PC		REL4G	REL3A		
FDBF	B, PC		REL4D	REL3A REL11H		
FORMAT	B, PC		REL2A	RELSOPL	SYMPL	
F45	B, PC		REL4F	REL3A		
IMF1	B, PC	SRT5	REL11B	REL3A REL11H		
LCS3	PC		REL5B			LCS3 does not install binaries.
NAM5	B, PC	NOTRACE	REL12G	RELSOPL		

†In the common keywords column, B indicates that the BINARY keyword is allowed and PC indicates that permanent code (PC keyword), corrective code (CC keyword), and site code (USERF keyword) are all allowed for the named procedure. Refer to Installation Procedure Parameters in this section for common keywords. Refer to the corresponding procedure name in section 5 for a description of unique keywords in this table.

NAM5 procedure:

The installation of the Network Access Method (NAM) performs the following tasks.

- Installs the following NAM components and utilities.

Mnemonic	Meaning
NAMI	Network Initialization Program
NIP	Network Interface Program
PIP	Peripheral Interface Program
NS	Network Supervisor
CS	Communications Supervisor
NVF	Network Validation Facility
DLFP	Debug Log File Processor
AIP	Application Interface Program routines residing on the NETIO and NETIOD libraries
QTRM	Queued Terminal Record Manager residing on the NETIO and NETIOD libraries
TVF	Terminal Verification Facility
NDLP	Network Definition Language Processor
LFG	CCP Load File Generator
NDA	NPU Dump Analyzer
COLLECT	Collect network dumps
LISTPPM	Format PIP memory dumps

- Retrieves from the program library on REL12G and saves the procedure files as indirect access permanent files under user name SYSTEMX (user index 377777_8) and NAMSTRT as an indirect access permanent file under user name NETOPSX (user index 377772_8).

Procedure Files	Network Initialization Program Startup Jobs
NAM NAMNOGO	NAMSTRT (contains startup jobs JOBNS, JOBCS, JOBTVF, JOBNVF, JOBNIP, JOBCOL, and JOBPUR; and parameter records INIT, RECOVR, and RESTRT)

The NDLP binaries must be in the running system or in GLOBLIB before creating the network and local configuration files, and the LFG binaries must be in GLOBLIB before creating the CCPLOAD file. If the current NDLP binaries are not in the running system but are in GLOBLIB, ~~execute the following procedure before calling NDLP.~~ SEE ERRATA

BEGIN,ATTGLOB,INSTALL.

The NAM1/CCP3 Reference Manual describes NIP, PIP, NS, CS, AIP, QTRM, TVF, and DLFP. The Network Definition Language Reference Manual describes NDLP. The NOS 2 System Maintenance Reference Manual describes LFG, NAMI, COLLECT, and NDA. Installation parameters for NAM are in section 5. If you install CCP, there are interdependencies between NAM and CCP. These interdependencies are established in file USERBPS (refer to USERBPS - User Build Parameters File in section 3 before installing NAM).

The flow of supervisory and data messages through the network is traced by Application Interface Program (AIP) code, which creates log files of such messages. The data that the log files provide is invaluable in the analysis of error conditions in network installation or operation. Startup jobs JOBNS, JOBCS, JOBTVF, and JOBNVF save the log files as direct access permanent files. If the network terminates abnormally, permanent files containing binary dumps of the program and any additional dump output will also be created. When the network is restarted, JOBCOL collects all permanent files on tape and purges them. For network problems, this tape (with other support materials) should be included with all PSRs submitted for network products. A more detailed description of the log file capability is in the NAM1/CCP3 Reference Manual.

To disable log file creation for the NS, NVF, and CS utilities, specify the NOTRACE keyword on the call to the procedure for the NAM installation.

Create the local and network configuration files with NDLP. Define these files as direct access permanent files with the names specified in the master file parameter records. LCFFILE and NCFFILE are the file names specified in the released version of the master file. Refer to the Network Definition Language Reference Manual for the procedure to create the configuration files.

LCFFILE and NCFFILE must be installed under user name NETOPSX (user index 377772_8).

Ensure that the system control point facility is enabled by entering ENABLE,SCP. in the IPRDECK (refer to section 9).

INSTALL GROUP 3

Because a group 3 procedure (table 2-4) requires the output binaries from a group 2 procedure as its input, the group 2 procedure must be completed before the corresponding group 3 procedure can be started. Refer to Install Group 1 for descriptions of table columns.

Table 2-4. Group 3 Products (Sheet 1 of 2)

Procedure Name	Keywords† Common	Keywords† Unique	Input Tape VSN	Other Tapes Required	Last Job That Used The Input Tape	Comments
BIT8	B, PC		REL3A		CCL	
CCP††	PC					Install after NAM.
CROSS††	PC					Install after NAM.
DCAT2	PC		REL11A			Install after COBOL5 or COBOL5Q. DCAT2 does not install binaries.
FCS3	PC		REL5B		LCS3	FCS3 does not install binaries.
FORM	B, PC		REL3A		BIT8	
FTN5	B, PC		REL4E	REL3A REL14B		Install after CCG.
MCS	B, PC	TRACE	REL12F	REL12G RELSOPL		Install after NAM.
PASCAL	B, PC		REL8C	RELSOPL		REQUIRES BINARY FIRST
PLI	B, PC		REL14A	REL14B REL3A		Install after CCG.
PMD4	B, PC		REL4C	REL3A	FCL2	
PMD5	B, PC		REL4G	REL3A	FCL5	
QU3	B, PC	SRT5, IMF	REL11E	REL11H RELSOPL		Install after DBU.††† If IMF is selected, install after IMF; if SRT5 is selected, install after Sort/Merge 5.
RBF5	B, PC	NOTRACE	REL12H	REL12G RELSOPL		Install after NAM.

†In the common keywords column, B indicates that the BINARY keyword is allowed and PC indicates that permanent code (PC keyword), corrective code (CC keyword), and site code (USERF keyword) are all allowed for the named procedure; and U indicates that only code from file USER is allowed. Refer to Installation Procedure Parameters in this section for common keywords. Refer to the corresponding procedure name in section 5 for a description of unique keywords in this table.

††Refer to section 3 for CCP and Cross installation information.

†††If Query Update 3 is to be used with DBU, DBU must be installed before QU3. If Query Update 3 is not to be used with DBU, and if DBU is not installed, unsatisfied external references cause error messages, but the operation of Query Update 3 is not affected.

Table 2-4. Group 3 Products (Sheet 2 of 2)

Procedure Name	Keywords† Common	Keywords† Unique	Input Tape VSN	Other Tapes Required	Last Job That Used The Input Tape	Comments
RHF	B, PC	SAVLCN	REL16A RELSOPL			REL16A required for binary installation; RELSOPL required for build.
RHP	B, PC		REL16B RELSOPL	REL16A		REL16B required for binary installation; RELSOPL required for build.
TAF	B, U	DEBUG TASK LIB	REL12C RELSOPL			REL12C required for binary installation; RELSOPL required for build.
IAF	B, U		REL12E RELSOPL			REL12E required for binary installation; RELSOPL required for build.
TOOLS	B, U		REL2A RELSOPL			REL2A required for binary installation; RELSOPL required for build.
MSS	B, U	SAVELIB	REL14C RELSOPL			REL14C required for binary installation; RELSOPL required for build.
TRACER	B, U		REL1G RELSOPL			REL1G required for binary installation; RELSOPL required for build.
HSIO	B, U		REL1J RELSOPL			REL1J required for binary installation; RELSOPL required for build.
GENSYS						Run after all group 3 procedures are completed. Refer to Step 3 - Create Deadstart Tape in this section.

†In the common keywords column, B indicates that the BINARY keyword is allowed and PC indicates that permanent code (PC keyword), corrective code (CC keyword), and site code (USERF keyword) are all allowed for the named procedure; and U indicates that only code from file USER is allowed. Refer to Installation Procedure Parameters in this section for common keywords. Refer to the corresponding procedure name in section 5 for a description of unique keywords in this table.

STEP 11 VERIFY INSTALLATION

You can verify the installation of products in two ways.

- Execute the REPORT procedure to get statistics on all completed installation procedures.

- Execute a verification procedure to determine if the installed product is operational.

The REPORT procedure can be executed after an installation job is completed.

The verification procedure can be executed after the product's binaries have been placed into the running system by the GENSYS procedure.

REPORT PROCEDURE

To obtain statistics on all completed installation jobs, run the following job. The job output indicates the resources used for each installation job and whether the job passed or failed.

```
job command.
USER,username,password,familyname.
CHARGE,*.
BEGIN,REPORT,INSTALL,XC.
--eoi—
```

NOTE

The report procedure uses the direct access file REP. If the binary file cannot be found or if the XC keyword is present, REP is recompiled.

VERIFICATION PROCEDURE SEE ERRATA

Check the installation of the products with the verification procedures (listed in table 2-5). A verification procedure can be run after the corresponding product is installed in the running system. Not all products have verification procedures. NAM and CCP have a different verification process (refer to Verify NAM and CCP in this section).

To verify that a product is successfully installed, run a job similar to the following.

```
job command.
USER,username,password,familyname.
CHARGE,*.
BEGIN,pname,INSTALL.
--eoi—
```

Parameter	Description
pname	pname is the verification procedure name of the product to be verified (refer to table 2-1).

The following job calls all verification procedures on file INSTALL except VCDCS2A, VCDCS2B, VMCS1A, and VMCS1B (refer to the descriptions of the CDCS2 and MCS procedures in section 5).

SEE ERRATA

```
job command.
USER,username,password,familyname.
CHARGE,*.
GET,USER=pfn.
BEGIN,VJOBS,INSTALL,D=density,PRID=printerid.
--eoi--
```

The parameter printerid can be specified to select any printer you require. All output files produced by VJOBS have a default identifier of 30 to facilitate printing on a single printer.

Refer to Installation Procedure Parameters in this section for a description of density.

Some verification procedures require that tapes be mounted; the operator will be requested to do this.

Since this job submits all verification procedures, procedures that verify products a site does not have will fail.

VERIFY NAM AND CCP

Use the following procedure to verify correct installation of NAM and CCP.

1. Master clear any local and remote network processing units (NPUs) (refer to the NOS 2 Operator/Analyst Handbook).

2. Initiate NAM at control point n by entering the following DSD command at the system console.

   ```
   ENABLE,NAM,n.
   NAM.
   ```

   If IAF is to be brought up, you must not initiate NAM at control point 1 (n must be 2 or greater).

When NAM actually starts execution (that is, after NAMI has completed execution), NAM will send its version number to the system dayfile. The message has the following format.

```
NAM VER ver - lvl
```

ver is the version number; lvl is the PSR summary level.

When the network spervisor (NS) starts execution, it places in NAM's dayfile its version number and the date and time that both the network configuration file (NCF) and the network load file (NLF) were built. These messages have the following format.

```
NS/    time VER ver - lvl.
NS/    time NCF build date, build time.
NS/    time NCF title.
NS/    time NLF build date, build time.
```

time is the time at which the message was sent to the dayfile.
ver is the product version number and lvl is the PSR summary level.
title is the site-supplied string to the network definition language processor (NDLP) for the NCF.

NS proceeds to load all accessible NPUs. NS issues load messages to the NAM dayfile (refer to the NOS 2 Operator/Analyst Handbook).

After the communications supervisor (CS) begins execution, it issues messages to the NAM dayfile having the following format.

```
CS/    time VER ver - lvl.
CS/    time NCF build date, build time.
CS/    time NCF title.
```

time is the time at which the message was sent to the dayfile.
ver is the product version number and lvl is the PSR summary level.
title is the site-supplied string to NDLP for the NCF.

As each NPU is loaded, CS issues messages to the NAM dayfile having the following format.

```
CS/    time NPU: npuname, AC npuid
CS/    time SUPERVISION GAINED
CS/    time CCP VERSION: ver, LEVEL: h, VARIANT: v
```

time is the time at which the message was sent to the dayfile.
npuname is the name of the NPU.
npuid is the node number of the NPU.
ver is the CCP version number.
h is a hexadecimal number indicating the level of code in this version of CCP (adjusted by Control Data; not an installation parameter).
v is a hexadecimal number identifying the CCP variant.

When the network validation facility (NVF) starts execution, it sends its version number and the local configuration file (LCF) build date and time to the NAM dayfile. These messages have the following format.

```
NV/    time VER ver - lvl.
NV/    time LCF build date, build time.
NV/    time LCF title.
```

time is the time at which the message was sent to the dayfile.
ver is the product version number and lvl is the PSR summary level.
title is the site-supplied string to NDLP for the LCF.

Table 2-5. Verification Procedures

Procedure Name	Input Tape VSN	Comments
VAAM2		
VALGOL5		
VAPL2		
VBAM		
VBASIC3		
VBIT8		
VCDCS2A		
VCDCS2B		
VCID		
VCOBOL5		
VCROSS	REL13A	
VDBU		
VDDL3		
VFCS3	REL5B	
VFDBF		
VFORM		
VFTN		
VFTNTS		
VFTN5		
VF45		
VIMF1		
VJOBS		Submits all verification jobs except VCDCS2A, VCDCS2B, VMCS1A, and VMCS1B.
VLCS3	REL5B	
VMCS1A		MCS must be enabled to execute VMCS1A (refer to section 5).
VMCS1B		MCS must be enabled to execute VMCS1B (refer to section 5).
VNPS2		
VPASCAL		
VPLI		
VQU3		Must be run as a batch origin job.
VSORT		
VSORT5		
VSYMPL		

After NS successfully loads an NPU, CS issues a message to the NAM dayfile for every logical link defined as active in the network configuration file. The message format is:

LLINK: linkname,EN,RL=3,H-N,npuname1/nid,npuname2/nid.

Parameter	Description
linkname	Name assigned to link in the network configuration file.
npuname1	Name of NPU that is connected to the coupler or trunk.
npuname2	Name of NPU at the other end of the logical link.
nid	End node number of logical link (either coupler or NPU node number).

NOTE

The remainder of this procedure determines that the network is running and able to process applications. TVF is used in this procedure; therefore, you must define TVF as an application (using an APPL statement) in the NDLP input file. TVF use is described in the NAM/CCP Reference Manual.

3. Assign the K display to NAM by typing the following commands at the system console.

```
K,NAM.
K.ST
```

The NAM status display should appear on the screen. A sample display appears as follows. Refer to the NOS 2 Operator/Analyst Handbook for an explanation of the format.

```
NIN = 1     REG LVL = 0     NO OF APPLS = 6   MAX-FL = 060000

APP   JSN     STATUS  I  NCN    AC    NSM    NDM    TIME UP
NVF   AAAU    000000  N  000          000    000    07.00.00
CS    AAAV    100000  N  000          000    000    07.00.01
NS    AAAW    100000  N  000          000    000    07.00.02
TVF   AAAX    100000     000          000    000    07.00.02
IAF   IAF     000000     000          000    000    07.00.04
RBF   RBF     100000     000          000    000    07.00.10

EST     HN   NSM    NHM    NLM     IVTSTAT    PRUST    NPUREJ
70       6   000                    0030       000       000
71       7   000                    0050       000       000

LOG-LINK    HN    TN    H  N  S  T  NCN    AC    NHDQ  NLDQ    TIME UP
            6     16    3  3  S     000                        07.01.00
            7     17    3  3  S     000                        07.01.02
```

All applications connected to NAM should appear in the applications section (the column titled APP) of the status display. All front end NPUs that are logically on in the EST should appear in the EST section of the display. All enabled/active logical links should appear in the logical link section of the display.

4. Log in from any network-supported terminal, specifying TVF as the application. The user name under which the login is performed must have permission to access TVF (refer to MODVAL in the NOS 2 System Maintenance Reference Manual). The NOS 2 Network Terminal User's Instant, and the NOS 2 Reference Set, Volume 3 describe the login procedures. TVF responds with several lines of information about the terminal followed by a prompt for input (..).

5. Enter 2 followed by the message transmission key for the terminal class in use (carriage return for most asynchronous terminals). This initiates the TVF line test. TVF responds with:

```
LINE TEST BEGINS
..
```

6. Enter any character followed by the message transmission key. TVF responds by printing a single line composed of the character you entered followed by:

```
TVF TEST COMPLETE
..
```

7. Enter END to exit TVF. The system responds by issuing a message indicating the time of connection to TVF followed by a prompt for application selection.

```
TVF      CONNECT TIME hh.mm.ss
termname    -APPLICATION:
```

8. Enter BYE to exit the network and log off. The system responds:

```
LOGGED OUT.
```

STEP 12 INSTALL CONTROLWARE

Install new controlware, if any (refer to section 4).

STEP 13 MOVE FILES

Some of the files created during the installation must be put in the appropriate permanent file catalog. The following products have files that must be relocated.

Product	Installation Procedure Name
Communications Control Program (CCP)	CCPLOAD
Conversion Aids System	LCS3, FCS3
~~Message Control System~~	~~MCS~~
Network Access Method	NAM5
~~Remote Batch Facility~~	~~RBF5~~
CYBRLOG	SYSJOB
~~Interactive Facility~~	~~IAF~~
~~Mass Storage Subsystem~~	~~MSS~~
~~Transaction Facility~~	~~TAF~~
XEDIT	XEDIT

The files to be moved, except those for CCP and CYBRLOG, are described under the installation procedure name in section 5. The files to be moved for CCP are described in section 3 and for CYBRLOG in table 2-2.

The MOVEPF utility moves files to a new user index; you must run it at the system console. The following format of MOVEPF is a DSD entry.

X.MOVEPF(UI=ui,DI=di,F1=pfn_1,F2=pfn_2,F3=pfn_3)

Parameter	Description
UI=ui	User index under which the pfn_i are stored.
DI=di	User index to which the pfn_i are being moved; 377777 indicates the system user index.
Fi=pfn_i	Name of permanent file to be moved. You can specify from one to three files.

The procedures needed to initiate a subsystem can exist either as a PROC type record on an indirect access file under user name SYSTEMX (UI = 377777) or as a PROC type record on the deadstart file with a corresponding *PROC directive in the LIBDECK.

The installation procedures place the following procedures on the deadstart file which, if necessary, can be altered and placed on SYSTEMX or replaced on the deadstart file.

Subsystem	Procedure Name
Interactive Facility	IAF, IAFTR, IAFTM
Transaction Facility	TAFPRC
Mass Storage Subsystem	MSS
MAGNET	MAG
Network Products	NAM, NAMNOGO

These procedures are not automatically installed as permanent files by the installation procedures. If they must be altered, use GTR to extract them from the deadstart file or use MODIFY to extract them from RELSOPL or the NAM PL and place them on permanent files before using the MOVEPF utility to place them under SYSTEMX.

Subsystem initiation procedures which exist as indirect access permanent files under SYSTEMX override corresponding procedures with the same name on the deadstart file.

FILES USED IN INSTALLATION PROCESS

The following is a partial list of files used during the installation and modification of the products. For a complete listing of these files and other installation procedures, run the job described in Step 8 - Set Up Installation Files in this section. Refer to CCP/Cross Permanent Files in section 3 for a detailed description of the files involved in the installation of Cross and CCP.

File	Description
CAPL	Direct access file containing permanent and corrective code for APL2.
CNSP	Direct access file containing permanent and corrective code for network products.

File	Description
CPRD	Direct access file containing permanent and corrective code for other optional products.
DAYFILS	Direct access file on which the dayfile for each installed product is kept.
DECKOPL	Modify-formatted program library that contains the procedures for installing the products.
GLOBLIB	Direct access file containing binaries of compilers, assemblers, and text that are used during the building of the system and products.
INSTALL	Direct access file ~~in user library format~~ containing all the installation procedures. For more information, obtain a listing of DECKOPL. SEE ERRATA
JOBSTAT	Direct access file on which the statistics information is kept for later processing by REPORT.
MDYMODS	File on RELO; it contains corrective code, which applies to the operating system and optional products that are maintained in Modify format. If corrective code is released, it must be installed. Refer to the CC parameter under Installation Procedure Parameters in this section.
MISCPL	File on RELO; it is copied to disk during the setup of installation files. This program library, which is in Update format, contains modsets that were available at system release but have not been fully tested. This code should not be installed in your normal build. You should install a modset from this file only after you have run with the released system and have encountered a problem that matches the description of a modset listed in this file.
PRODUCT	Direct access file containing binaries of various products. Installation procedures for various products add appropriate binaries to this file via the LIBEDIT utility.
REP	Direct access file containing the binary of the report-generating program REPORT.
RELSOPL	Modify-formatted source program library containing the source code for the operating system and optional products. The COMBINE installation procedure creates this program library from tape REL1A and from the REL tapes containing the code for the selected optional products. RELSOPL may be a magnetic tape or a permanent file depending upon the installation parameter DISKINS.
USER	Local file, established by the installer, which contains code modifications for a product.

USER LIBRARIES

The operating system and product installation procedures on RELO create certain user libraries. The following user libraries are created if all the products supported by NOS are installed (these library names are reserved for Control Data).

Library Name	Associated Product Name	Installation Procedure Name
AAMLIB	Advanced Access Methods	AAM2
ALG5LIB	ALGOL-60	ALGOL5
BAMLIB	Basic Access Method	BAM
BASLIB	BASIC	BASIC3
BCLIB	Transaction Facility	TAF
BIT8LIB	8-Bit Subroutines	BIT8
COBOL5	COBOL 5	COBOL5
DBUGLIB	CYBER Interactive Debug	CID
DMSLIB	Data Definition Language	DDL3
FORTRAN	FORTRAN Extended 4	FTN
FTN5LIB	FORTRAN 5	FTN5
IMFLIB	Information Management Facility	IMF1
LCNLIB	Remote Host Facility	RHF
NETIO	Network Access Method	NAM5
NETIOD	Network Access Method	NAM5
PASCLIB	PASCAL 170	PASCAL
PLILIB	PL/I	PLI
SRTLIB	Sort/Merge 4	SORT
SRT5LIB	Sort/Merge 5	SORT5
SRVLIB	NOS	-
SYMLIB	SYMPL	SYMPL
SYSLIB	Many products	COMBINE
TRANC5	Transaction Facility	TAF
TRANF4	Transaction Facility	TAF
TRANF5	Transaction Facility	TAF
TRANLIB	Transaction Facility	TAF

To obtain catalogs of these object libraries, run the following job for each library listing you want (the library can be cataloged only after the product with which it is associated is installed).

Job	Comment
job command.	
USER,username,password,familyname.	
CHARGE,*.	
COMMON,SYSTEM.	
GTR,SYSTEM,LIB,D.ULIB/libname	libname is the name of the object library.
CATALOG,LIB,N,R,U.	
--eoi--	

The user name specified on the USER command must have permission to access library files.

INSTALLATION OF CYBER CROSS SYSTEM AND COMMUNICATIONS CONTROL PROGRAM — 3

Install the CYBER Cross System and Communications Control Program (CCP) after the Network Access Method (NAM), Interactive Facility (IAF), Remote Batch Facility (RBF), Transaction Facility (TAF), and Message Control System (MCS).

HARDWARE REQUIREMENTS

A field length of 110000_8 is required to build CCP.

The following equipment configuration is the minimum required to execute CCP:

- One 2550-2 or 2551-1 Host Communication Processor, consisting of:

  One multiplexer loop interface adapter.
  One loop multiplexer.
  One cyclic encoder board.
  One CYBER communications coupler.
  One 32K memory unit.
  One 8K micromemory board.

- One communications line adapter (CLA), either a 2560-1 synchronous CLA or a 2561 asynchronous CLA.

- Additional memory of 48K.

Assign the communications line adapter slots in the loop multiplexer in order of decreasing line transmission speeds. For example:

Speed	Slot Assignment
9600-bps line	Slot 1 (leftmost slot)
9600-bps line	Slot 2
2400-bps line	Slot 3
300-bps line	Slot 4
150-bps line	Slot 5

BUILD STEPS DESCRIPTION

The CCP/Cross installation procedures consist of seven sequential build steps. If only Cross will be installed, do only the first build step (CROSS). The following description of the build steps lists them in their proper execution sequence.

Build Step	Description
CROSS	Updates the Cross program library on the REL13A tape with corrective code from file CPRD and with user corrective code from file UCRS; compiles the updated binaries for use by the CCP build steps; writes an updated version of tape REL13A. If you will not install CCP, skip the remaining build steps.
CCPPH1	Updates the program libraries on the REL13B, REL13E, and REL13F tapes with corrective code from file CNSP and then merges the updated program libraries into file PCMB; creates updated program libraries of CCP (PCCP), of on-line diagnostics (PDGN), and of remote concentrator products (PREM); updates PCMB with temporary user-supplied corrective code from file UCCP and generates the phase 1 (micromemory) and dump load modules on file ZMUX; creates EXPAND and Autolink binaries; builds system autostart (SAM) load module; writes updated versions of tapes REL13E and/or REL13F.
CCPBLB	Updates the PCMB program library with temporary user-supplied corrective code from file UCCP and generates the CCP object code library (BCMB); writes an updated version of REL13B. This build step is also called the CCP full compile and assembly build step.
CCPVAR	Updates the PCMB program library with temporary user-supplied corrective code from file UCCP and generates a CCP variant load module (Zvvv) and program initiation control block (PICB) file (Ivvv) from the BCMB file according to the user-specified variant definitions in file USERBPS; writes tape REL13C. This build step should be repeated for each NPU in the network. If this build step is repeated in separate batch jobs, assign a different REL13C tape for each build step. This prevents overwriting the information written by the previous CCPVAR build.
CCPEDIT	Patches a CCP variant load module. This build step is not part of the normal build process but allows the use of the MPEDIT utility of Cross.
CCPLOAD	Updates the PCMB program library with temporary user-supplied corrective code from file UCCP and generates a NAM network load file (Gzzz) via program LFG (refer to the NOS 2 System Maintenance Reference Manual). The load file includes the phase 1 and dump load modules (file ZMUX) and system autostart load module (file ZSAM) from step CCPPH1, and the variant load modules ($Zvvv_i$) and PICBs (Ivvv) from step CCPVAR.

Build Step	Description
CCPPURG	Purges the noncritical permanent files created by the other build steps. It does not purge the load file from build step CCPLOAD and the user-supplied files. This build step is not required; it is only a cleanup utility. However, since previous build steps do not purge the noncritical permanent files, it is suggested that CCPPURG be run to make more disk space available.

The final result of the CCP/Cross build steps is the generation of a NAM network load file. Figure 3-1 illustrates the build step dependencies. Figure 3-2 illustrates the relationship of the load file to the release tapes and the other files critical to the CCP build process. Figure 3-3 illustrates the relationship of the build steps to the critical files and tapes involved in CCP installation.

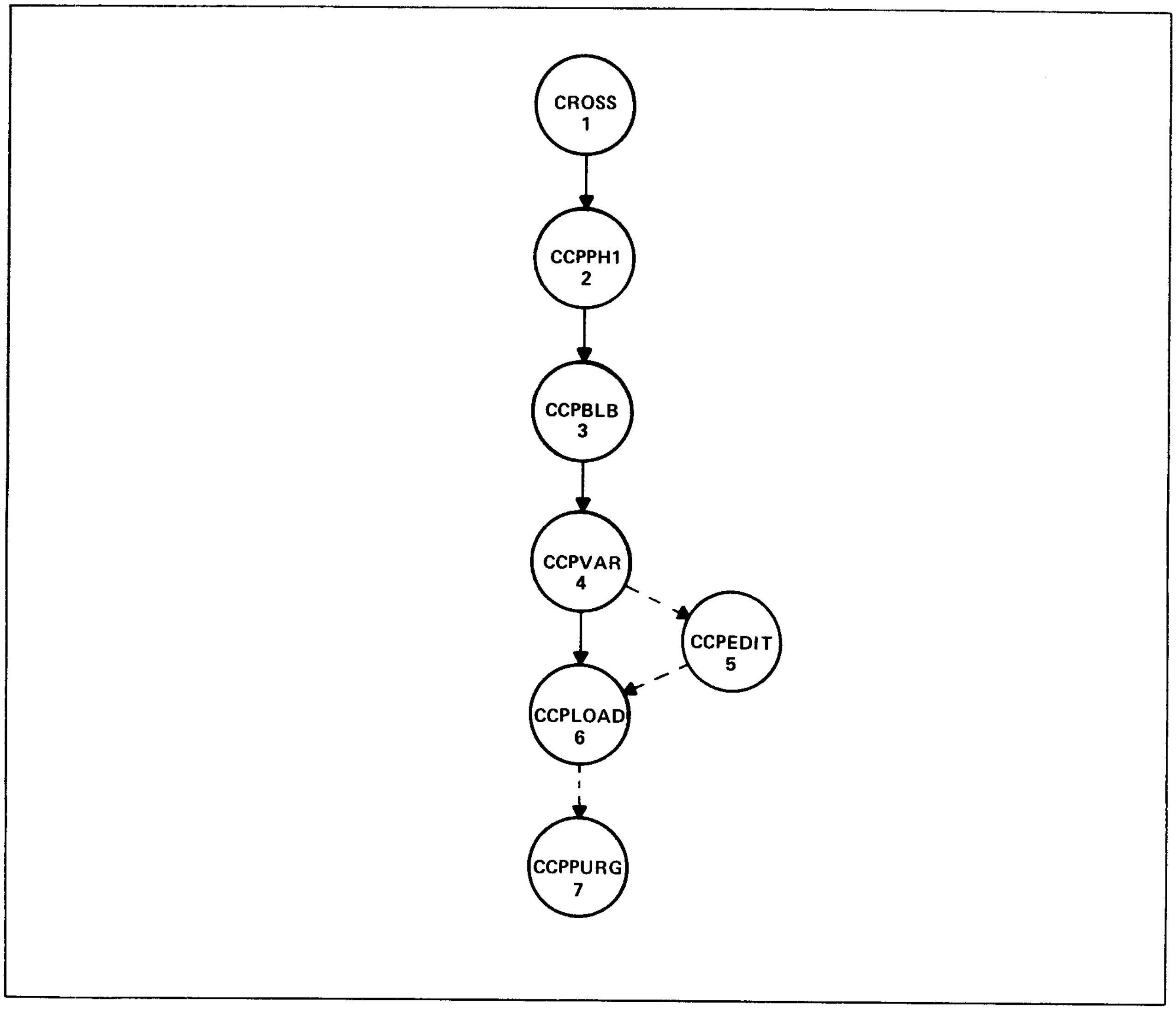

Figure 3-1. CCP/Cross Build Step Dependencies

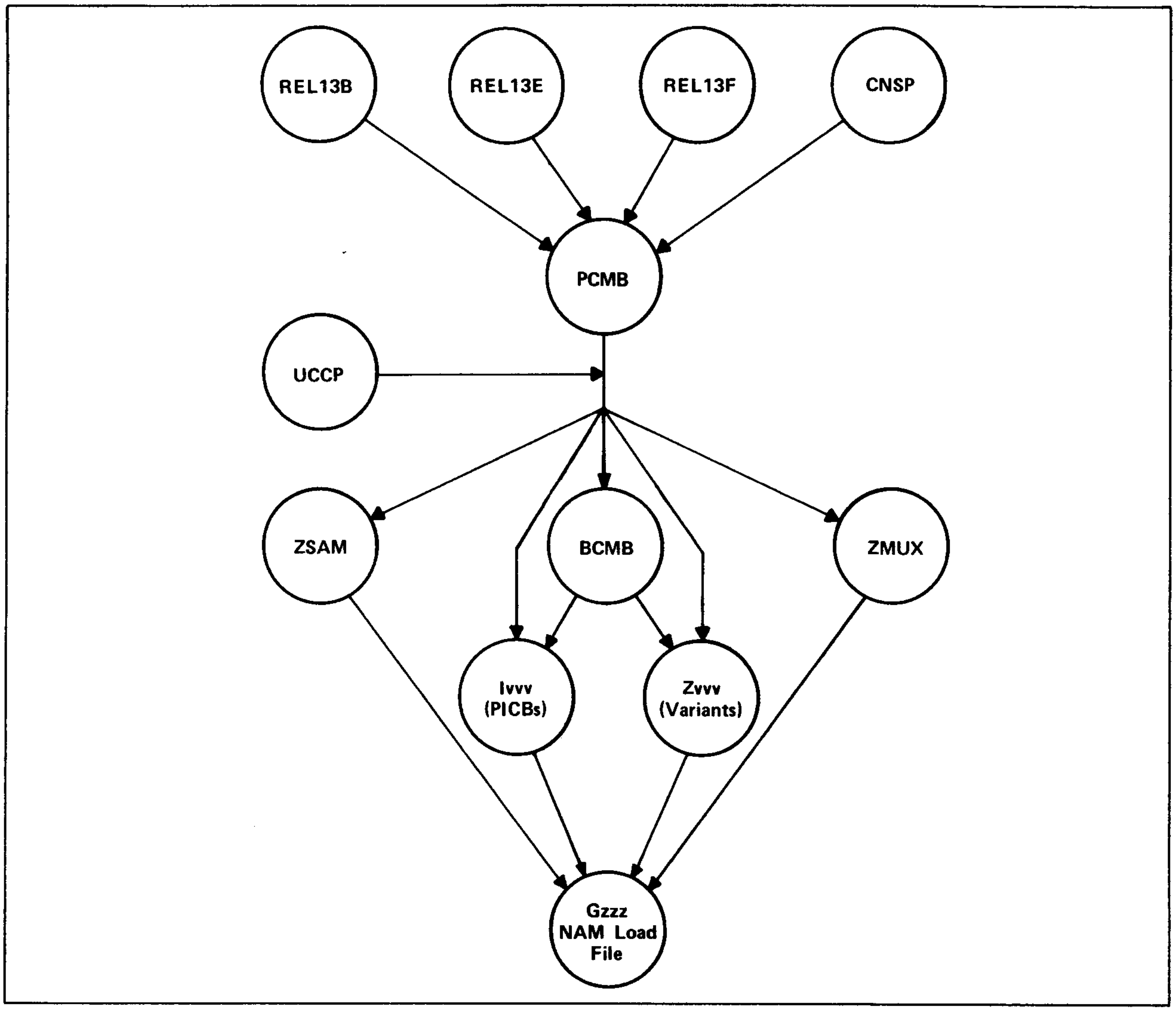

Figure 3-2. CCP File Dependencies

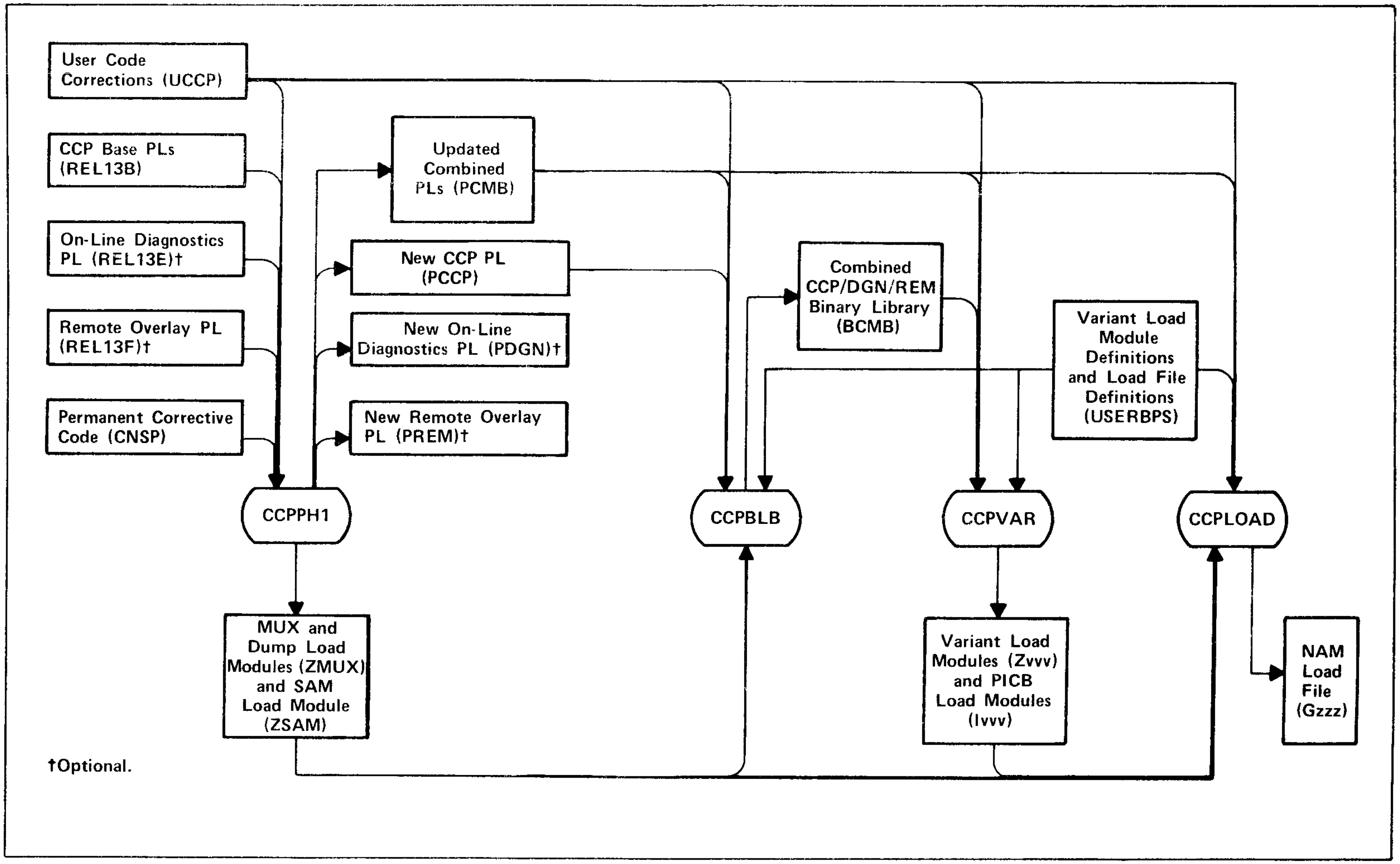

Figure 3-3. Integration of Program Libraries in CCP Build Process

GENERAL BUILD STEP CALL

All CCP/Cross build steps are called by the BEGIN command. Descriptions of each of the seven sequential build step procedures, including the required BEGIN parameters, are in subsequent subsections. Table 3-1 summarizes the tape and disk file requirements of the build steps.

Table 3-1. CCP/Cross Tape and Disk File Requirements

Build Step Order	Build Step Name	Input Tape VSN	Output Tape VSN	Input Files Generated by Previous Step	User Input Files	Permanent Files Created	Optional Permanent Files Created
1	CROSS	REL13A	REL13A		UCRS USERCHG CPRD	Addd PCRS	LCRB
2	CCPPH1	REL13B REL13E REL13F	REL13E REL13F		UCCP USERCHG CNSP	ZMUX PCMB PCCP ZSAM Addd SMUX	LIMC LMFB PDGN PREM LSAM
3	CCPBLB		REL13B	PCMB LSAM LIMC PCCP ZMUX LMFB ZSAM SMUX Addd	UCCP USERCHG USERBPS	BCMB	LFCA
4	CCPVAR		REL13C	PCMB BCMB SMUX	UCCP USERBPS USERCHG	Zvvv Svvv Ivvv	Lvvv
5	CCPEDIT (optional)			Zvvv Svvv	UEDZ USERCHG	Zyyy Syyy	
6	CCPLOAD			PCMB ZMUX ZSAM Zvvv Ivvv	UCCP USERBPS USERCHG	Gzzz	
7	CCPPURG (optional)				USERCHG		

To use the 63-character set for CCP/Cross, the 63-character set modification parameter must have been specified in the call to procedure SETUP. Refer to SETUP Procedure in section 2.

The format of the BEGIN command is:

BEGIN,pname,INSTALL,p_1,p_2,...,p_n.

Parameter	Description
pname	Name of the build step procedure (refer to table 3-1).
p_i	Build step parameter. The order-independent format should be used.

p_i	Description
CC=cc	Specifies whether build step uses corrective code from CNSP or CPRD. If this parameter is omitted, CC=YES is assumed.

cc	Description
YES	Corrective code from CNSP or CPRD is used. YES is only required when there is corrective code on the RELO tape.
NO	Corrective code from CNSP or CPRD is not used.

p_i	Description
D=density	Specifies density of release tape(s) used in build step; must be GE, HY, or PE. If this parameter is omitted, D=PE is assumed.

density	Description
GE	6250 cpi.
HY	800 cpi.
PE	1600 cpi.

p_i	Description
DIAG=diag	Specifies whether on-line diagnostics are present (REL13E); used only with CCPPH1. If this parameter is omitted, DIAG=NO is assumed.

diag	Description
YES	On-line diagnostics are present.
NO	On-line diagnostics are not present.

p_i	Description
GN=file	Specifies load file name. The user supplies the three-character, alphanumeric file name; used only with CCPLOAD.

Parameter	Description

P_i	Description
LIST=option	Specifies whether the build step creates a listing, saves the listing as a permanent file on disk, and/or assigns the listing to OUTPUT. Do not specify this parameter for the CCPLOAD build step. If this parameter is omitted on the CCPVAR build step, LIST=PF is assumed. If this parameter is omitted on the CROSS, CCPPH1, or CCPBLB build step, LIST=NO is assumed.

option	Description
YES	Listing is assigned to OUTPUT. For the CCPVAR build step, listing is assigned to OUTPUT and is copied to the new release tape.
PF	Listing is stored as a permanent file on disk; later it is copied to the new release tape and purged from the disk.
BOTH	Listing is stored as a permanent file as well as assigned to OUTPUT; later it is copied to the new release tape and the permanent file is purged.
NO	No listing is created.

NEW=yyy	Specifies new CCP variant name for patched load module; used only with CCPEDIT. Supply the three-character alphanumeric name.
NOECS	Specifies for build steps CCPPH1 and CCPBLB that extended memory will not be used. Supply this parameter only when extended memory is down or when you do not want to use extended memory even if it is available. Do not supply this parameter if there is no extended memory or if fewer than 77000_8 words of extended memory are available.
OLD=xxx	Specifies CCP variant to be patched; used only with CCPEDIT. Supply the three-character alphanumeric name.

Parameter	Description

P_i	Description
PC=pc	Controls the inclusion of code to update the released program libraries from the previous release to the PSR summary level of the current release. If the product program library was included with the current installation materials, PC=NO should be used. The default is PC=NO.

pc	Significance
YES	Updates program library to current PSR summary level.
NO	Does not update the release program library.

R=type

SEE ERRATA

Specifies the type of auxiliary device that will store the permanent files. type consists of three characters. The first two are alphabetic characters specifying the device type. The third character is a number within the range 1 through 8; it identifies the number of units composing the device. For example, the parameter R=DI2 specifies two 844 units to be accessed as a single logical device. If the third character is omitted from type, 1 is assumed. If this parameter is omitted, the system default device type is assumed.

type	Equipment
DB†	885-42, 7155-401 (full track).
DI	844-21, 7054/7154 (half-track).
DJ	844-41/44, 7054/7154 (half-track), 7155.
DK	844-21, 7154 (full-track).
DL	844-41/44, 7154 (full-track), 7155.
DM	885-11/12, 7155 (half-track).
DQ	885-11/12, 7155 (full-track).

This parameter assumes either the presence of a PACKNAM command in USERCHG or that a PACKNAM command precedes the build step in the installation job.

SEE ERRATA

†Not applicable for models 815, 825, 835, and 855.

Parameter	Description

P_i	Description
REMT=remt	Specifies whether remote concentrator products are present (REL13F); used only with CCPPH1. If this parameter is omitted, REMT=NO is assumed.

remt	Description
YES	Remote concentrator products are present.
NO	Remote concentrator products are absent.

VN=vvv	Specifies a variant name that matches the variant name in the VRD definition in USERBPS; used only with CCPVAR.
Vx=vvv	Specifies the variant name that was used in CCPVAR; used only with CCPPURG. x can be an integer within the range 1 through 10.
XREF=xref	Specifies whether the build step generates a cross-reference listing of the Pascal source of CCP; used only with CCPBLB. If this parameter is omitted, XREF=NO is assumed.

xref	Description
YES	The cross-reference listing is generated.
NO	The cross-reference listing is not generated.

CROSS – CROSS SYSTEM INSTALLATION

The following build step generates updated program binaries for all Cross programs and installs those programs needed for the following CCP build steps. Refer to General Build Step in this section for descriptions of the parameters.

```
BEGIN,CROSS,INSTALL,D=density,R=type,LIST=option,PC=pc.
```

The CROSS build step uses the following files for input.

File	Description
USERCHG	User/charge file. It is optional for all build steps (refer to CCP/Cross Permanent Files in this section).
REL13A	Cross release tape.
UCRS	Optional site corrective code (refer to CCP/Cross Permanent Files in this section). For a description of the Cross installation parameters that can be changed, refer to Installation Parameters in this subsection.

File	Description
CPRD	Cross corrective code, if any, that affects the resulting Cross binaries but is not placed in the program library on the output tape.

The CROSS build step creates the following output files.

File	Description
NEW13A	New REL13A tape (VSN of REL13A).
LCRB	Cross system listings (if requested).
APAS	MP17 Pascal compiler.
AASM	Cross macro assembler.
AMAC	Macro assembler text file.
AMAS	Cross micro assembler.
AFMT	Pascal binary output formatter program.
AXRF	Pascal cross-reference program.
ALNK	MPLINK program.
AEDT	MPEDIT program.
ALIB	MPLIB program.
ACYP	CYBER 170, CYBER 70, and 6000 Computer Systems Pascal compiler.

Examples:

You can execute the CROSS build step by using any of the following methods.

```
BEGIN,CROSS,INSTALL.
```

By default, this command sets the track and density of REL13A to be nine-track and 1600 cpi, respectively. The system device is the default device type. The build step generates no listings and installs the corrective code.

```
BEGIN,CROSS,INSTALL,D=HY,LIST=YES.
```

This command sets the track and density of REL13A to be seven-track and 800 cpi, respectively. The CROSS build step copies its listings (LCRB) to OUTPUT. By default, the system device is the default device type and the build step installs the corrective code.

```
BEGIN,CROSS,INSTALL,R=DJ1,LIST=PF,PC=NO.
```

By default, this command sets the track and density of REL13A to be nine-track and 1600 cpi, respectively. LCRB is stored for the duration of the job on the auxiliary pack of device type DJ1. The corrective code on file CPRD (Update-formatted) makes corrections to the release tape REL13A.

Hardware Requirement

The CROSS build step requires a field length of 110000_8.

Installation Parameters

Identifier	Parameter	Released Default Value	Significance
XSYA127.6	MAXGLBL	1535	Maximum number of global symbols minus one.
XSYA127.7	HGHPAGE	55	(SYMTBSIZ/32)-1.
XSYA127.8	SYMTBSIZ	1792	Size of in-core symbol table.
XSYA127.9	VARPAGE	47	MAXGLBL/32.
XSYA127.406	SYMTBSIZ	1792	Size of in-core symbol table.
XSYA127.407	MAXGLBL	1535	Maximum number of global symbols minus one.

The number of entries in the in-core symbol table in the release version of the Pascal compiler is 1792. This version of the compiler has a corresponding maximum number of global symbol definitions of 1536 and an execution field length of 77000_8 central memory (CM) words. Some programs require a Pascal compiler that accommodates more than 1536 global symbol definitions; for example, CCP requires 6144 global symbols. Increasing the size of the global symbol table without increasing the in-core symbol table, however, results in a significant increase in compilation time. Further, an increase in the number of CM words must accompany any increase in the size of the in-core symbol table (four CM words per symbol table entry).

CCPPH1 - CCP PHASE 1

The following build step generates a combined base program library for CCP, on-line diagnostics, and remote concentrator program libraries. It also creates the multiplexer (MUX) firmware and the dump load module. Refer to General Build Step in this section for descriptions of the parameters.

```
BEGIN,CCPPH1,INSTALL,D=density,R=type,LIST=option,DIAG=diag,REMT=remt,CC=cc,PC=pc,NOECS.
```

The CCPPH1 build step uses the following input files.

File	Description
USERCHG	User/charge file. Refer to User-Supplied Files in CCP/Cross Permanent Files in this section.
UCCP	Optional user corrective code. Refer to User-Supplied Files in CCP/Cross Permanent Files in this section.
REL13B	CCP release tape.
REL13E	Diagnostic release tape (if DIAG=YES).

File	Description
REL13F	Remote release tape (if REMT=YES).
CNSP	CCP corrective code.

CCPPH1 generates the following output files.

File	Description
PCMB	Updated combined program library.
LMFB	CCP list file. File 1: MUX firmware. File 2: dump bootstrap overlay.
LSAM	System autostart (SAM) listing (if requested).
ZMUX	CCP load module. File 1: MUX firmware. File 2: dump bootstrap overlay.
PCCP	New CCP program library including corrective code from CNSP.
PDGN	New diagnostic program library including corrective code from CNSP.
PREM	New remote program library including corrective code from CNSP.
SMUX	Dump bootstrap symbol table.
ZSAM	System autostart (SAM) load module.
AEXP	Build parameters expand program binary.
AALK	Autolink program binary.
LIMC	Listing of Expand and Autolink programs.

Examples:

You can execute the CCPPH1 build step in several different ways, including:

BEGIN,CCPPH1,INSTALL,DIAG=YES,REMT=YES.

DIAG=YES and REMT=YES signify that both on-line diagnostics and the remote dump/load overlay are to be installed. This command selects the defaults for density, auxiliary pack device type, and corrective code use.

BEGIN,CCPPH1,INSTALL,DIAG=YES.

DIAG=YES signifies the presence of the on-line diagnostics. This command selects the defaults for density, auxiliary pack device type, listings, remote concentrator products, and corrective code use.

```
BEGIN,CCPPH1,INSTALL,LIST=YES,REMT=YES.
```

LIST=YES automatically routes the list files to the printer. REMT=YES signifies the presence of the remote dump/load overlay. This command selects the defaults for density, auxiliary pack device type, diagnostics, and corrective code use.

CCPBLB – CCP BINARY LIBRARY

The following build step generates an updated combined binary library of all CCP procedures and assembly language subroutines. Refer to General Build Step in this section for descriptions of the parameters.

```
BEGIN,CCPBLB,INSTALL,D=density,R=type,LIST=option,XREF=xref,NOECS.
```

CCPBLB requires the following input files.

File	Description
USERCHG	User/charge file. Refer to User-Supplied Files in CCP/Cross Permanent Files in this section.
USERBPS	User variant build parameters file. Refer to User-Supplied Files in CCP/Cross Permanent Files in this section.
UCCP	Optional user corrective code. Refer to User-Supplied Files in CCP/Cross Permanent Files in this section.
PCMB	Updated combined program library.
PCCP	New CCP program library including corrective code from CNSP.
ZMUX	CCP load module. File 1: MUX firmware. File 2: dump bootstrap overlay.
LMFB	CCP list file. File 1: MUX firmware. File 2: dump bootstrap overlay.
AEXP	Build parameters expand program binary.
AALK	Autolink program binary.
LIMC	Listing of expand and autolink programs.
LSAM	System autostart (SAM) listing.
SMUX	Dump bootstrap symbol table.
ZSAM	System autostart (SAM) load module.

CCPBLB produces these output files.

File	Description
NEW13B	CCP release tape (VSN of REL13B).
BCMB	Combined CCP/diagnostics/remote binary library.
LFCA	Full compile assembly listings. File 1: Assembly source listing. File 2: Pascal source and object listing.

The CCPBLB build step must follow CCPPH1 and requires a field length of 77000_8.

Example:

BEGIN,CCPBLB,INSTALL,LIST=YES,XREF=YES.

This command selects the defaults for density and auxiliary pack device type. LIST=YES routes the listings generated by CCPBLB to the printer and does not make them a permanent file. XREF=YES causes the XREF program to generate a cross-reference listing of the Pascal source of CCP.

CCPVAR – CCP VARIANT

The following build step generates a CCP variant (phase 2) load module and a PICB load module based on user-supplied variant definitions on file USERBPS. Refer to General Build Step in this section for descriptions of parameters.

BEGIN,CCPVAR,INSTALL,D=density,R=type,LIST=option,VN=vvv.

CCPVAR requires the following input files.

File	Description
USERCHG	User/charge file. Refer to User-Supplied Files in CCP/Cross Permanent Files in this section.
UCCP	Optional user corrective code. Refer to User-Supplied Files in CCP/Cross Permanent Files in this section.
PCMB	Updated combined program library.
BCMB	Combined CCP/diagnostics/remote binary library.
USERBPS	User variant build parameters file. Refer to User-Supplied Files in CCP/ Cross Permanent Files in this section.
SMUX	Dump bootstrap symbol table from CCPPH1 build step.

CCPVAR generates the following output files.

File	Description
NEW13C	Variant release tape (VSN of REL13C). If the CCPVAR build step is repeated in separate batch jobs, a different REL13C tape must be assigned for each job to prevent overwriting the previous REL13C file.
Zvvv	CCP variant load module (vvv is the variant name).
Ivvv	PICB load module.
Svvv	Symbol table for CCP variant load module Zvvv and PICB load module Ivvv.
Lvvv	Variant load module and PICB listing.

CCPVAR requires a field length of 110000_8.

Example:

BEGIN,CCPVAR,INSTALL,VN=FEP.

This command selects the defaults for track, density, auxiliary pack device type, and listings. VN=FEP creates the files ZFEP, IFEP, SFEP, and LFEP (a load module, a PICB, a symbol table, and a variant link edit listing respectively).

CCPEDIT — CCP LOAD MODULE FILE EDIT

The following build step patches an absolute CCPLOAD module (file named Zvvv, where vvv is the CCP variant load module) via a special MPEDIT run (refer to CYBER Cross System Build Utilities Reference Manual). The CCP build process requires this step only for those cases where there is a minor difference between an existing load module and the desired load module. Refer to General Build Step in this section for descriptions of the parameters.

BEGIN,CCPEDIT,INSTALL,OLD=vvv_1,NEW=vvv_2,R=type.

CCPEDIT requires four input files.

File	Description
USERCHG	User/charge file. Refer to User-Supplied Files in CCP/Cross Permanent Files in this section.
$Zvvv_1$	CCP variant load module vvv_1.
UEDZ	Optional direct or indirect access permanent file of MPEDIT directives to patch a CCP variant load module. Refer to User-Supplied Files in CCP/Cross Permanent Files in this section.
$Svvv_1$	Symbol table associated with variant load module vvv_1.

CCPEDIT produces two output files.

File	Description
$Zvvv_2$	New CCP variant load module reflecting patch code.
$Svvv_2$	Copy of symbol table $Svvv_1$.

Example:

BEGIN,CCPEDIT,INSTALL,OLD=FEP,NEW=FE2.

CCPEDIT patches variant FEP with MPEDIT patch code directives from UEDZ to create a new variant FE2 and selects the default for the auxiliary pack device type.

CCPLOAD – GENERATE CCP LOAD FILE

Based on user-supplied load file definitions on file USERBPS, the following build step generates a CCP load file used by NAM/NS to downline load NPUs. Refer to General Build Step in this section for a description of the parameter.

BEGIN,CCPLOAD,INSTALL,R=type,GN=zzz.

CCPLOAD requires these input files.

File	Description
PCMB	Updated combined program library. Refer to User-Supplied Files in CCP/ Cross Permanent Files in this section.
USERCHG	User/charge file. Refer to User-Supplied Files in CCP/Cross Permanent Files in this section.
UCCP	Optional user corrective code. Refer to User-Supplied Files in CCP/Cross Permanent Files in this section.
USERBPS	CCP load file definitions file supplied by user. Refer to User-Supplied Files in CCP/Cross Permanent Files in this section.
ZMUX	MUX firmware and dump bootstrap load module.
ZSAM	System utostart load module.
Zvvv	Previously created CCP variant load module(s).
Ivvv	Previously created PICB load module(s).

CCPLOAD generates one output file.

File	Description
Gzzz	CCP load file (zzz is the value associated with the GN keyword).

Example:

```
BEGIN,CCPLOAD,INSTALL,GN=XYZ.
```

This command uses the load file definitions on file USERBPS and creates the CCP load file GXYZ used by NAM/NS. It selects the default for the auxiliary pack device type.

Relocate file:

If the released version of the NS job skeleton JOBNS is used (refer to NAM5 - Network Access Method Version 1, in section 5), rename Gzzz file as NLFFILE and move NLFFILE to NETOPSX (user index 377772_8).

CCPPURG – CCP/CROSS INSTALLATION FILES PURGE

The following optional build step purges all noncritical permanent files created by the CCP/Cross installation process. This step does not purge the user-supplied files and the CCP load file created by CCPLOAD. Refer to General Build Step in this section for descriptions of parameters.

BEGIN,CCPPURG,INSTALL,R=type,V1=vvv_1,V2=vvv_2,...,Vn=vvv_n.

CCPPURG requires one input file.

File	Description
USERCHG	User/charge file. Refer to User-Supplied Files in CCP/Cross Permanent Files in this section.

This step produces no output files.

Example:

```
BEGIN,CCPPURG,INSTALL,V1=FEP,V2=LOC,V3=REM,V4=NP1,V5=NP2.
```

This command selects the default for the auxiliary pack device type and purges all permanent files associated with the variants FEP, LOC, REM, NP1, and NP2. A CATLIST after this procedure verifies that the system has purged all the permanent files created by the installation process. CCPPURG does not purge the files USERCHG, USERBPS, UCCP, UCRS, UEDZ, and the load file (Gzzz) created by CCPLOAD.

CCP/CROSS PERMANENT FILES

All permanent files generated by the CCP/Cross installation procedures are named by the following convention: each name consists of four characters and the first character identifies the file type. The first character can be any of the following:

File Type	Description
A	Absolute load file (Cross program).
B	Binary library or LGO file.
C	Permanent corrective code in Update format with master control character of /.
G	CCP load file created by the load file generator (LFG) program.
I	Program initiation control block (PICB).
L	CCP/Cross listing (generated during installation).
P	Program library in Update format.
S	CCP symbol table.
U	User supplied corrective code file.
Z	Load modules required by CCPLOAD.

An alphabetical list of permanent files generated by the CCP/Cross installation follows. The files are grouped by their file type.

Absolute Load Files	Description
AALK	Autolink program.
AASM	Cross macro assembler.
ACYP	CYBER 170, CYBER 70, and 6000 Computer Systems Pascal compiler.
AEDT	MPEDIT program.
AEXP	Build parameters expand program.
AFMT	Pascal binary output formatter program.
ALIB	MPLIB program.
ALNK	MPLINK program.
AMAC	Macro assembler text file.

Absolute Load Files	Description
AMAS	Cross micro assembler.
APAS	MP17 Pascal compiler.
AXRF	Pascal cross-reference program.

Binary Library File	Description
BCMB	Combined CCP/diagnostics/remote binary library.

Corrective Code Files	Description
CPRD	Product corrective code (on RELO).
CNSP	CCP/Network Host Products corrective code (on RELO).

CCP Load File	Description
Gzzz	CCP load file generated by CCPLOAD (the zzz appended to the letter G is the value of the GN parameter).

CCP/PICB Load Modules	Description
Ivvv	CCP program initiation control block load modules.

CCP/Cross Listings	Description
LCRB	Cross system listings.
LDGB	Diagnostics build listing.
LFCA	Full compile assembly listings.
LIMC	Expand and autolink program listings.
LMFB	MUX firmware and dump bootstrap overlay listings.
LRMB	Remote overlay build listing.
Lvvv	Variant load module listing (vvv is variant name).

Program Libraries	Description
PCCP	New CCP program library including corrective code on CNSP.
PCMB	Updated combined program library.
PCRS	New Cross program library including corrective code on CPRD.
PDGN	New diagnostic program library including corrective code on CNSP.
PREM	New remote program library including corrective code on CNSP.

Symbol Tables	Description
SMUX	Symbol table for dump bootstrap.
Svvv	Symbol table for load module Zvvv.

NOTE

All CCP/Cross user-supplied files must be permanent files under the same user name used for the build step jobs. The USERBPS and USERCHG files must be indirect access permanent files. The UCRS, UCCP, and UEDZ files can be indirect or direct access permanent files; local files of the same name are ignored. Table 3-1 summarizes the tape and disk file requirements of the build steps.

User-Supplied Files	Description
UCCP	Optional direct or indirect access permanent file of user code corrections to CCP. The contents of this file should be the same for all build steps requiring it.
UCRS	Optional direct or indirect access permanent file of user code corrections to Cross. This file may be used only with build step CROSS.
UEDZ	Optional direct or indirect access permanent file of MPEDIT directives to patch a CCP variant load module. This file may be used only with build step CCPEDIT.

User-Supplied Files	Description
USERBPS	User build parameters file. This indirect access permanent file contains the CCP system definition, the CCP variant load module definitions, and the CCP load file definitions. This file is required for build steps CCPBLB, CCPVAR, and CCPLOAD. For each execution of CCPVAR, the USERBPS file must remain unchanged. A complete description of USERBPS immediately follows the listing of the permanent files.
USERCHG	User/charge file. This indirect access permanent file contains the USER command, the CHARGE command (if required), and commands that are executed at the start of the build step. If the user specifies the R=type parameter in a build step, this file must also contain a PACKNAM command. This file is required for all build steps done from a terminal. If a CCP/Cross installation procedure is executed from a batch job, omit the USERCHG file.

Load Modules	Description
ZMUX	MUX load module firmware and dump bootstrap overlay.
ZSAM	System autostart (SAM) load module.
Zvvv	CCP variant load module (vvv is variant name).

NOTE

If the CCP/Cross build process is interrupted, you must ensure that the required files are present upon resumption.

USERBPS file:

Create a build parameters file (indirect access permanent file USERBPS) containing a CCP system definition, CCP variant load module definitions, and CCP load file definitions. Build steps CCPBLB, CCPVAR, and CCPLOAD require this file. During build step CCPBLB, the utility program EXPAND searches through USERBPS for the system (SYS) definition. It then expands the definition, according to a macro text file, into Update directives that control the options and TIPS that are assembled and compiled into the combined binary library (BCMB). For build steps CCPVAR and CCPLOAD, parameters specify the desired variant or load file definition. EXPAND then searches for and expands the definition in the same manner as described for the system definition. The Update directives created cause input to be generated for the AUTOLINK program.

USERBPS can contain any number of CCP system, variant, and load file definitions. If more than one system definition is present, only the first definition will be used. The format of CCP build definitions is:

$keyword_1=value_1,keyword_2=value_2,\ldots,keyword_n=value_n.$

When a keyword takes on multiple values, the form is:

$keyword_1=value_1/value_2/\ldots/value_n.$

This is equivalent to:

$keyword_1=value_1,keyword_1=value_2,\ldots,keyword_1=value_n.$

The following syntax rules apply to all definitions.

- The first keyword must be one that identifies the type of definition (VRD indicates a variant definition and LFD indicates a load file definition).
- EXPAND ignores all embedded blanks. Blank lines are illegal.
- A period terminates each definition.
- Continuation lines must begin with a plus (+).
- EXPAND treats any line whose first character is an asterisk (*) as a comment line.
- When a definition takes more than one line, the user should break the definition between parameter pairs.

NOTE

Several parameters in the USERBPS definitions are interdependent with parameters in the CMRDECK entries and in the Network Definition Language (NDL) statements. These parameters are noted in the following parameter descriptions. Additionally, when creating the network configuration file with the Network Definition Language Processor (NDLP), the NODE parameter in the HOST statement must have a value of 1, and the P1LID parameter in the NPU statements must have a value of MIC (refer to the Network Definition Language Reference Manual).

CCP SYSTEM DEFINITION

The system definition controls the options and terminal interface programs (TIPS) that are assembled and compiled into the combined binary library (BCMB). It is similar to the variant definition (described in the following subsection), but must include all options and all TIPS that will be used in any variants to be built from the resulting combined binary library.

The system definition can continue over more than one line as long as each line prior to the last ends with a comma. The last line must end with a period. The system definition has two parts, either of which may be present or absent. The resulting four formats are as follows.

Format	Significance
SYS.	No options, no TIPs.
SYS=<options>.	Options present, no TIPs.
SYS,TS=<TIPs>.	TIPs present, no options.
SYS=<options>,TS=<TIPs>.	Both options and TIPs present.

Keyword	Description
SYS=v1/v2/v3	Specifies options if present.

vi	Description
C	Support modules for CONSOLE (for printing CCP information on a terminal connected to a 2550) will be compiled.
D	Online diagnostic support modules are present.
P	Support modules for statistics on line/trunk/ NPU performance, which are logged on the account dayfile and the error log file, will be compiled.
R	Remote concentrator products are present.
T	Support modules for TUP† (test utility program) and CONSOLE will be compiled.

Keyword	Description
TS=$t_1/t_2/.../t_n$	Specifies the TIPs that are to be included in the system. TS can assume up to 10 different order-independent values.

t_i	Description
A	Asynchronous TIP is included. This TIP supports ASCII terminals, APL character sets, and IBM 2741 terminals.
B	Binary synchronous communications (BSC) TIP is included. This TIP supports the IBM 2780 and IBM 3780 terminals.
H	HASP TIP is included.
M	Mode 4 TIP is included.
XP	X.25 packet assembly/disassembly (PAD) TIP is included. Specify this TIP for any variant that executes in an NPU connected to a packet switching network.

†TUP is an unsupported product.

t_i	Description
XA	X.25 application-to-application TIP is included.
1	User TIP1 is included.
2	User TIP2 is included.
3	User TIP3 is included.
4	User TIP4 is included.

The following example includes all the options and TIPs specified in the examples shown for the variant load module definition.

```
SYS=R/D/T,TS=A/B/H/M/XP/XA.
```

CCP VARIANT LOAD MODULE DEFINITION

The variant load module definition can continue over more than one line as long as each line ends with comma, except the last line must end with a period.

The format is:

VRD=vvv,VT=$v_1/v_2/v_3$,SZ=xK,TS=$t_1/t_2/.../t_n$.

Keyword	Description
VRD=vvv	Identifies entry as a variant definition and specifies variant name (associated vvv value). Build step CCPVAR uses vvv to create unique permanent file names. Specify a three-character alphanumeric string, beginning with an alphabetic character. It must not be the same as the last three characters of the CCP/Cross permanent file names (refer to CCP/Cross Permanent File Names).
VT=$v_1/v_2/v_3$	Specifies variant type of the NPU. You can associate a maximum of three separate values with VT. One of the following values must appear.

v_i	Description
F	Front-end; includes Host Interface Program (HIP) but no Local Interface Program (LIP).
L	Local; includes HIP and LIP.
R	Remote; includes LIP but no HIP.

Keyword	Description
	The following values are optional.

v_i	Description
C	Variant includes module CONSOLE.
D	Variant includes on-line diagnostic support modules.
P	Variant includes modules for statistics/ performance results.
T	Variant includes modules TUP and CONSOLE for debugging.

Examples:

VT=L/D/T

VT=F/D

VT=R

Keyword	Description
SZ=xK	Specifies variant memory size: 65K, 81K, 96K, or 128K (x is a two- or three-digit number).
TS=$t_1/t_2/\ldots/t_n$	Specifies which Terminal Interface Programs (TIPs) are to be included in this variant. TS can assume up to 10 different order-independent values.

t_i	Description
A	Asynchronous TIP is included. This TIP supports the ASCII terminals, APL character sets and IBM 2741 terminals.
B	Binary synchronous communications (BSC) TIP is included. This TIP supports the IBM 2780 and IBM 3780 terminals.
H	HASP TIP is included.
M	Mode 4 TIP is included.
XP	X.25 packet assembly/disassembly (PAD) TIP is included. Specify this TIP for any variant that executes in an NPU connected to a packet switching network.
XA	X.25 application-to-application TIP is included.
1	User TIP1 is included.
2	User TIP2 is included.

t_i	Description
3	User TIP3 is included.
4	User TIP4 is included.

Example 1:

VRD=EX1,VT=L/D,SZ=81K,TS=A/M.

This variant supports an 81K local NPU with asynchronous and mode 4 TIPs and on-line diagnostics.

Example 2:

VRD=EX2,VT=R/T,SZ=96K,TS=A/H/XP.

This variant supports a 96K remote NPU with HASP, X.25 PAD, and asynchronous TIPs. This variant does not support on-line diagnostics but supports a 2550 console.

Example 3:

VRD=EX3,VT=F/D/T,SZ=128K,TS=A/B/H/M/XP/XA.

This variant supports a 128K front-end NPU with no remote NPUs, all TIPs (except site-defined TIPs), and on-line diagnostics. This variant supports a 2550 console.

CCP LOAD FILE DEFINITION

The format is:

LFD=zzz,LM=$vvv_1/vvv_2/.../vvv_n$.

Keyword	Description
LFD	Identifies entry as a load file definition and specifies the last three characters of the load file name (associated zzz value). The zzz value must be a three-character alphanumeric string matching the corresponding GN=zzz parameter in the build step CCPLOAD. CCPLOAD uses this value to create a unique permanent file name for the output file. zzz must not be the same as the last three characters of any of the CCP/Cross permanent file names (refer to CCP/Cross Permanent File Names).
LM	Specifies the CCP variant load modules and PICB load modules to include in this load file. The MUX firmware (phase 1), dump load, dump bootstrap, and SAM modules are automatically included in every load file. The associated value vvv_i is the three-character name of a variant load module (file name $Zvvv_i$) that was generated by the CCPVAR build step. Repeat the vvv_i specification (separated by slants) for each variant to be included in the load file.

Example 1:

```
LFD=EX4,LM=EX1/EX2/EX3.
```

This entry defines a load file containing the variants created in the three CCP variant definition examples.

Example 2:

```
LFD=EX5,LM=EX3.
```

This entry defines a load file containing only the variant in the third CCP variant definition example.

CCP/CROSS INSTALLATION EXAMPLES

Examples follow which illustrate installation of CCP in two network configurations, one NPU and three NPUs (two local NPUs and one remote NPU).

- Both examples require release tapes REL13A and REL13B (example 2 also requires REL13E and REL13F).
- Both examples require the following input files.

Files	Description
USERCHG	Required for all build steps; contains USER and CHARGE commands.
UCCP	Required for user-suggested or PSR code for CCP.
USERBPS	Required for CCPBLB, CCPVAR and CCPLOAD; contains CCP system definitions, variant definitions and load file definitions.
UCRS	Required for user-suggested or PSR code for Cross.

  Refer to table 3-1 for a complete list of tape and file requirements for the build steps.

- In the build steps, both examples use the defaults for density, auxiliary pack device type, and inclusion/exclusion of corrective code.
- In both examples, underlined and lettered parameters indicate the interdependence among USERBPS definitions, CMRDECK entries, NDL source input, and build steps. Parameters with the same letter must match within each example.

Example 1: One NPU

The configuration includes the following.

- 81K NPU.
- Mode 4 TIP.
- Asynchronous TIP.

- No remote node support software.
- No NPU console support.
- No on-line diagnostics.

The following procedure illustrates the installation of CCP with a single NPU.

1. Ensure that the required files and tapes are available. Refer to figure 3-4 for appropriate USERBPS definitions, CMRDECK entries, and NDL source input.

2. Install Cross.

   ```
   BEGIN,CROSS,INSTALL.
   ```

   This step requires REL13A and a field length of 110K. Cross is needed for the CCP build process.

3. Build the phase (micro code and dump bootstrap) and SAM load modules.

   ```
   BEGIN,CCPPH1,INSTALL.
   ```

   This step requires REL13B.

4. Create an updated combined binary library of all CCP Pascal procedures and assembly language subroutines.

   ```
   BEGIN,CCPBLB,INSTALL,LIST=PF,XREF=YES.
   ```

   LIST=PF stores the listings on disk as permanent files during the build job. At the end of the job, the files are copied to the new release tape and purged. XREF=YES generates a Pascal cross-reference listing.

5. Create the phase 2 variant load module.

   ```
                                   b
   BEGIN,CCPVAR,INSTALL,LIST=PF,VN=VN1.
   ```

   The load module has a file name of ZVN1 and the PICB has a file name of IVN1. LIST=PF stores the listings on disk as permanent files during the build job. At the end of the job, the files are copied to the new release tape and purged.

6. Create the load file used by NAM/NS to downline load the NPU.

   ```
                              g
   BEGIN,CCPLOAD,INSTALL,GN=EX1.
   ```

   The load file name is GEX1.

```
*USERBPS
*ONE NPU EXAMPLE
*
*2550 WITH:     ASYNC TIP     NO ONLINE DIAGNOSTICS
*               MODE 4 TIP    NO REMOTE
*               81K MEMORY    NO NPU CONSOLE
*
*
*SYSTEM DEFINITION IS
*
*       a
SYS,TS=A/M.
*
*VARIANT DEFINITION IS
*
*    b                  a
VRD=VN1,VT=F,SZ=81K,TS=A/M
*
*LOAD FILE DEFINITION IS
*
*    g      b
LFD=EX1,LM=VN1.
```

USERBPS definition

```
NCF2P1:NFILE.
 d                         b
NP1:NPU  NODE=3, VARIANT=VN1.
                   c
  SUPLINK  LLNAME=LL23.
                   e
COUP2:COUPLER  NODE=2, HNAME=HOST1.
    c                   d
  LL23:LOGLINK  NCNAME=NP1.
END.
```

NDL Source Input

```
                 e
EQ41=NP,ON,7,1,5,,2,0.
```

CMRDECK Entry

Figure 3-4. USERBPS Definitions, NDL Source Input, and CMRDECK Entry for Example 1

Example 2: Three NPUs

Figure 3-5 illustrates the configuration of the network for this example. It shows the size of each NPU, the external connections (trunks, lines, and/or coupler) to each NPU, and the interface programs (TIPs and HIP and/or LIP) included in each NPU. It also shows the node number and port assignment(s) and/or NDL name for major components in the network as chosen for this example. In the configuration shown in figure 3-5:

- NPUA has three TIPs, a HIP, and a LIP. The latter two are required for the coupler and trunk, respectively.
- NPUB has two TIPs as well as a HIP and a LIP.
- NPUC has three TIPs and a LIP. A HIP is not required since no coupler is used.

NPUA and NPUC have on-line diagnostics; NPUC has console support. NPUC can communicate with the network through the remote node software of either NPUA or NPUB.

The following procedure illustrates the installation of CCP with a network configuration as shown in figure 3-5.

1. Ensure that the required files and tapes are available. Refer to figure 3-6 for appropriate USERBPS definitions, CMRDECK entries, and NDL source input.

2. Install Cross.

   BEGIN,CROSS,INSTALL.

   This step requires REL13A and a field length of 110K.

3. Build the phase 1 (micro code and dump bootstrap) and SAM load modules.

   BEGIN,CCPPH1,INSTALL,LIST=PF,DIAG=YES,REMT=YES.

   This step requires REL13B, REL13E, and REL13F. LIST=PF stores the listings on disk as permanent files; DIAG=YES specifies that on on-line diagnostics are present; REMT=YES specifies that remote concentrator products are present.

4. Create an updated combined binary library of all CCP Pascal procedures and assembly language subroutines.

   BEGIN,CCPBLB,INSTALL,LIST=BOTH,XREF=YES.

   This step requires REL13B. LIST=BOTH routes the listings to the printer and stores them as permanent files during the build job. At the end of the job, the files are copied to the new release tape and purged. XREF=YES generates a Pascal cross-reference listing.

5. Create the phase 2 variant load module for NPUA.

   BEGIN,CCPVAR,INSTALL,LIST=PF,VN=$\underline{\text{VNA}}^{a}$.

   LIST=PF stores the listings as permanent files during the build job. At the end of the job, the files are copied to the new release tape and purged. The load module file name is ZVNA, and the PICB file name is IVNA.

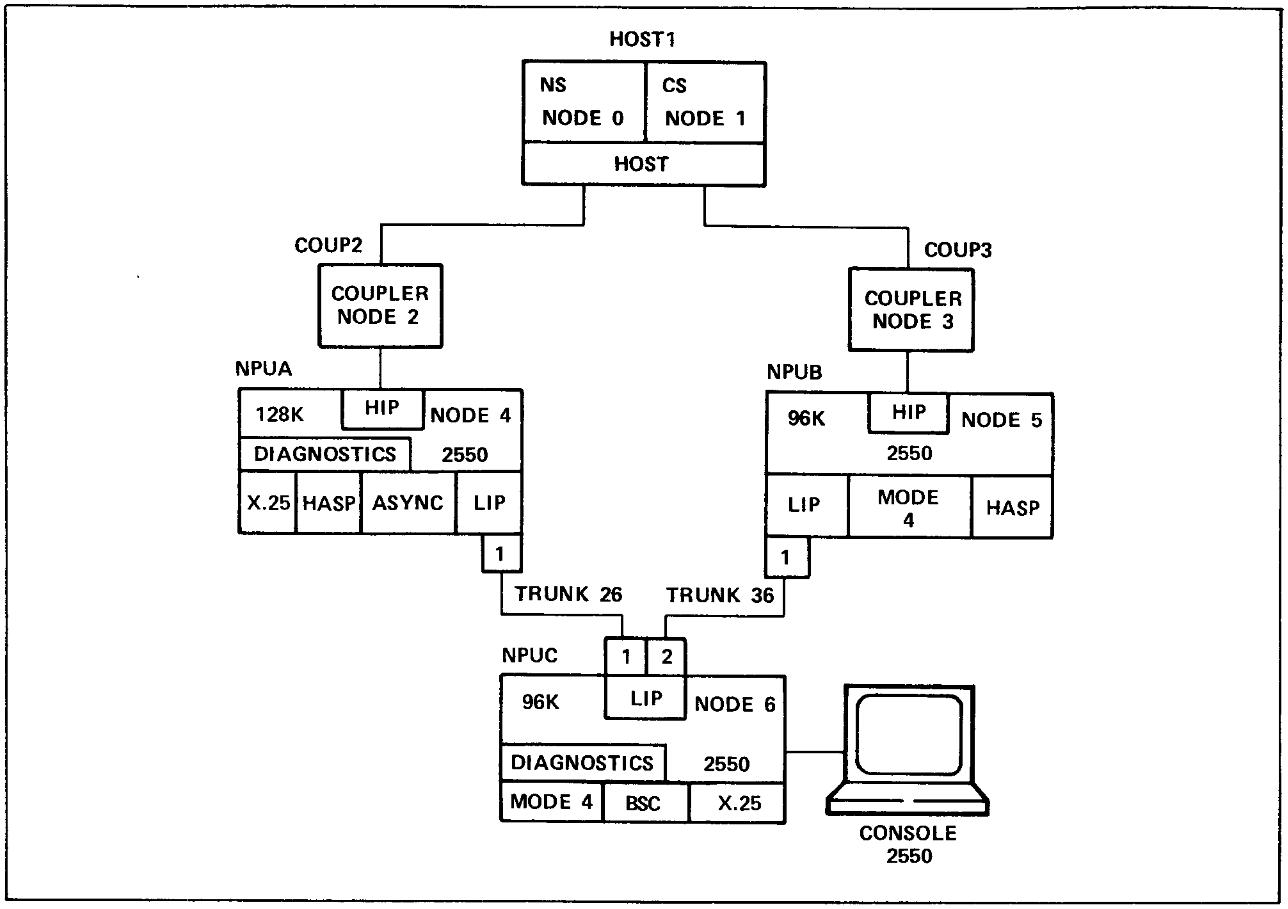

Figure 3-5. Network Configuration - Example 2

6. Create the phase 2 variant load module for NPUB.

```
                         b
   BEGIN,CCPVAR,INSTALL,VN=VNB.
```

   The load module file name is ZVNB, and the PICB file name is IVNB.

7. Create the phase 2 variant load module for NPUC.

```
                                 c
   BEGIN,CCPVAR,INSTALL,LIST=PF,VN=RMC.
```

   The load module file name is ZRMC and the PICB file name is IRMC. LIST=PF stores the listings as permanent files during the build job. At the end of the job, the files are copied to the new release tape and purged.

8. Create the load file used by NAM/NS to downline load the NPUs.

```
                          n
   BEGIN,CCPLOAD,INSTALL,GN=EX3.
```

   The load file name is GEX3.

```
*USERBPS
*THREE-NPU EXAMPLE
*
*SYSTEM DEFINITION IS
*
*   e d f    g h i j k
SYS=R/D/C,TS=A/B/H/M/XP.
*
*VARIANT DEFINITIONS ARE
*
*    a       d              g i k
VRD=VNA,VT=L/D,SZ=128K,TS=A/H/XP.
*    b                     j i
VRD=VNB,VT=L,SZ=96K,TS=M/H.
*    c     e d f            j h k
VRD=RMC,VT=R/D/C,SZ=96K,TS=M/B/XP.
*
*LOAD FILE DEFINITION IS
*
*    n      a   b   c
LFD=EX3,LM=VNA/VNB/RMC.
```

USERBPS
Definitions

```
NCF2P1: NFILE.
 l                        a
NPUA: NPU NODE=4, VARIANT=VNA.
                        m
     SUPLINK LLNAME=LL24.
                            o
     COUP2: COUPLER  NODE=2,HNAME=HOST1.
      m                     l
     LL24: LOGLINK NCNAME=NPUA.
      p                     q
     LL26: LOGLINK NCNAME=NPUC.

 r                        b
NPUB: NPU NODE=5, VARIANT=VNB.
                         s
     SUPLINK  LLNAME=LL35.
                           t
     COUP3: COUPLER NODE=3, HNAME=HOST1.
      s                     r
     LL35: LOGLINK NCNAME=NPUB.
      u                     q
     LL36: LOGLINK NCNAME=NPUC.
 q                        c
NPUC: NPU NODE=6, VARIANT=RMC.
                        p
     SUPLINK  LLNAME=LL26.
                        u
     SUPLINK  LLNAME=LL36.
                          l       q
     TRUNK26:TRUNK  N1=NPUA,N2=NPUC,P1=1,P2=1.
                          r       q
     TRUNK36:TRUNK  N1=NPUB,N2=NPUC,P1=1,P2=2.
END.
```

NDL
Source
INPUT

```
                    o
EQ41=NP,ON,7,1,5,,2,0.
                    t
EQ42=NP,ON,7,1,5,,3,0.
```

CMRDECK
Entries

Figure 3-6. USERBPS Definitions, NDL Source Input, and CMRDECK Entries for Example 2

9. Execute this build step only if you want to purge extraneous files.

```
                                a      b      c
BEGIN,CCPPURG,INSTALL,V1=VNA,V2=VNB,V3=RMC.
```

This step purges all extraneous permanent files associated with variant names VNA, VNB, and RMC.

CONTROLWARE INSTALLATION 4

Controlware for 844/885 mass storage and 667/669 magnetic tape devices is available on either punched cards or nine-track magnetic tapes. If you want the magnetic tape versions, specify magnetic tape when ordering the controlware: cards are supplied with the standard release materials. The 819 PPU driver is only available on magnetic tape.

If new controlware is not being installed, skip this section and skip the BCS, BCF, FMD, PHD, HCD, and MTS installation jobs in table 2-2. The following procedures install the 819 PPU driver from magnetic tape and the controlware from either punched cards or magnetic tape.

819 PPU DRIVER INSTALLATION

The 819 PPU driver can be installed only on a model 176. The 819 PPU driver resides on the system as a PPU-type record named HCD and is loaded into the first-level peripheral processors (FLPPs) during deadstart. The installation procedure assembles the 819 driver from the release tape and copies the binaries to the permanent file PRODUCT.

To build a system with a new 819 driver from release tape REL3A, submit a job similar to the following:

```
job command.
USER,username,password,familyname.
CHARGE,*.
BEGIN,HCD,INSTALL.
--eoi--
```

844/885 CONTROLWARE INSTALLATION

The 844/885 controlware resides on the system as a PPU-type record named BCS for half-track 7054/7152/7154 controlware, BCF for full-track 7152/7154 controlware, FMD for 7155-1 controlware, or PHD for 7155-401 controlware; it is loaded into the 7054, 7152, 7154, 7155-1 or 7155-401 controller during system deadstart. To build a system with new controlware from either punched cards or magnetic tape, follow one of the following procedures. The procedures copy the 844/885 controlware program from the released deck or tape to the permanent file PRODUCT.

PUNCHED CARDS

Submit the following job to install 844/885 controlware from cards.

Job	Comments
job command.	
USER,username,password,familyname.	
CHARGE,*.	
NOTE,IN.+*COMMENT PPU/typ nn	typ is the controlware type. BCS Half-track 7054/7152/7154. BCF Full-track 7152/7154. FMD 7155-1. PHD 7155-401. nn is the version of 844/885 controlware you are installing.
COPYBF,INPUT,INHOLD.	
BEGIN,typ,INSTALL.	typ is the controlware type (previously defined).
--eor--	
844/885 controlware deck.	
--eoi--	

MAGNETIC TAPE

To install 844/885 controlware from magnetic tape, submit a job similar to the following.

Job	Comments
job command.	
USER,username,password,familyname.	
CHARGE,*.	
LABEL,TAPE,F=SI,NT,D=1600,LB=KU.	Before entering this command, the 844/885 controlware tape should be mounted and ready on a nine-track tape drive. The tape must be assigned by the operator.
NOTE,IN.+*COMMENT PPU/typ nn	typ is the controlware type. BCS Half-track 7054/7152/7154. BCF Full-track 7152/7154. FMD 7155-1. PHD 7155-401. nn is the version of 844/885 controlware you are installing.
COPYBF,TAPE,INHOLD.	
RETURN,TAPE.	
BEGIN,typ,INSTALL.	typ is controlware type (previously defined).
--eoi--	

667/669 CONTROLWARE INSTALLATION

The 667/669 controlware resides on the system as a PPU type record named FIRM66X and is loaded into the 7021 or 7152 controller at system deadstart and whenever the magnetic tape subsystem, MAG, is brought to a control point. To build a system with new controlware from either punched cards or nine-track magnetic tape, follow one of the following procedures. The procedures copy the 667/669 controlware program from the released deck or tape to the permanent file PRODUCT.

PUNCHED CARDS

Figure 4-1 illustrates the structure of the coldstart deck used to load the 667/669 controlware when deadstarting from 667 or 669 magnetic tape units (coldstart).

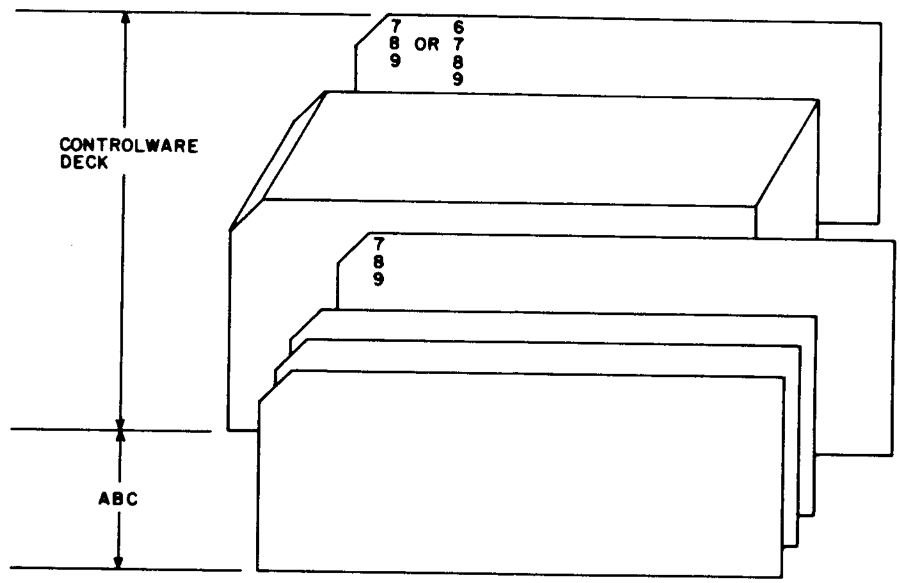

Figure 4-1. Coldstart Deck Structure

There are two records in the coldstart deck, the automatic buffer controlware loader (ABC) program and the controlware deck. The program on the deadstart panel reads and executes ABC, which causes the controlware deck to be loaded. When controlware loading is complete, system or maintenance deadstart proceeds.

NOTE

ABC is not on the deadstart tape, but is on the REL2A tape.

Use the following job to add new controlware programs to the deadstart tape and also to obtain the coldstart deck. It collects the binary card deck for the 667/669 magnetic tape controlware program FIRM66X, installs the binaries on the permanent files used to create a new deadstart tape, and punches a coldstart deck. (This coldstart deck is also supplied with the released controlware.)

Job	Comments
job command.	
USER,username,password,familyname.	
CHARGE,*.	
NOTE,IN.+*COMMENT PPU/FIRM66X nn	nn is the version of 667/699 controlware being installed.
COPYBF,INPUT,INHOLD.	
BEGIN,MTS,INSTALL.	
--eor--	
667/669 controlware deck.	
--eoi--	

MAGNETIC TAPE

To install 667/669 controlware from magnetic tape, enter a job similar to the following.

Job	Comments
job command.	
USER,username,password,familyname.	
CHARGE,*.	
LABEL,MTSTAPE,F=L,NT,D=1600,LB=KU.	Before entering this command, the 667/669 controlware tape should be mounted and ready on a nine-track drive. The tape must be assigned by the operator.
NOTE,IN.+*COMMENT PPU/FIRM66X nn	nn is the version of the 667/669 controlware being installed.
SKIPF,MTSTAPE.	
COPY,MTSTAPE,TEMP,TC=EOF,N=1,BS=10000B.	
RETURN,MTSTAPE.	
REWIND,TEMP.	
COPYBR,TEMP,INHOLD.	
BEGIN,MTS,INSTALL.	
RETURN,PUNCHB.	
--eoi--	

NETWORK ACCESS DEVICE (NAD) CONTROLWARE INSTALLATION

The Network Access Device (NAD) controlware resides on the system as a PPU-type record named 170 for NADs connected to CYBER 170 computer systems. Controlware named 200 for NADs connected to CYBER 200 computer systems, IBM for NADs connected to IBM systems, and MIN for NADs connected to minicomputer (DEC) systems may also be installed on the system as a PPU-type record. It is loaded into the NADs during RHF initialization or on operator request. To build a system with new 170 NAD controlware or to install IBM, DEC, or 200 controlware from either punched cards or magnetic tape, follow one of the following procedures. The procedures copy the NAD controlware program from the released deck or tape to the permanent file PRODUCT.

PUNCHED CARDS

Submit the following job to install NAD controlware from cards.

Job	Comments
job command.	
USER,username,password,familyname.	
CHARGE,*.	
NOTE,IN.+*COMMENT PPU/typ nn	typ is the controlware type. 170 CYBER 170 NADs. 200 CYBER 200 NADs. IBM IBM NADs. MIN Minicomputer (DEC) NADs. nn is the version of NAD controlware you are installing.
COPYBF,INPUT,INHOLD.	
BEGIN,typ,INSTALL.	typ is the controlware type (previously defined).
--eor--	
NAD controlware deck.	
--eoi--	

MAGNETIC TAPE

To install NAD controlware from magnetic tape, submit a job similar to the following.

Job	Comments
job command.	
USER,username,password,familyname.	
CHARGE,*.	
LABEL,TAPE,F=SI,NT,D=1600,LB=KU.	Before entering this command, the NAD controlware tape should be mounted and ready on a nine-track tape drive. The tape must be assigned by the operator.
NOTE,IN.+*COMMENT PPU/typ nn	typ is the controlware type. 170 CYBER 170 NADs. 200 CYBER 200 NADs. IBM IBM NADs. MIN Minicomputer (DEC) NADs. nn is the version of NAD controlware you are installing.
COPYBF,TAPE,INHOLD.	
RETURN,TAPE.	
BEGIN,typ,INSTALL.	typ is controlware type (previously defined).
--eoi--	

CONTROLWARE CARD DECK GENERATION

To punch a controlware card deck, run the following job.

Job	Comments
job command.	
USER,username,password,familyname.	
CHARGE,*.	
COMMON,SYSTEM.	
GTR,SYSTEM,FILE.PPU/dname	dname is the name of the record containing the controlware desired.
ATTACH,OPL=RELSOPL.	If RELSOPL is on tape, use a LABEL command.
MODIFY,C,Z./*EDIT ABC	
COMPASS,I,S=NOSTEXT.	
COPYBR,FILE,LGO.	
ROUTE,LGO,DC=SB.	
--eoi--	

Extract the first card from the resulting card deck before inserting the deck in subsequent jobs.

IDENTIFICATION PROCEDURE FOR 844/885 AND 667/669 CONTROLWARE

If it is necessary to determine which controlware level is available, do one of the following:

- Examine the last two data cards in the card deck (refer to table 4-1).
- Examine the specific memory locations in the controller's high core, using the buffer controller maintenance console (refer to table 4-1).
- Use the GTR command to get the desired record from the deadstart tape and the CATALOG command on the record to see the controlware level.

When examining the cards in a card deck, note that each hexadecimal word of the controller's core memory is punched in two successive columns (refer to table 4-1, Examine Columns on Card Deck). The higher order character is first in each pair of columns. On each card, columns 1 and 2 contain word count and checksum, column 76 is unused, and columns 79 and 80 contain a sequence number.

In table 4-1, the figures in parentheses apply to a controlware card deck for the 885.

Table 4-1. Location of Controlware Identification

Octal Card	Examine Columns on Card Deck	Examine Memory Locations Contained in Columns	Portion of Controlware Level
156 (341)	14, 15 (71, 72)	0FFD (17C8)	Product identification (hexadecimal equivalent of 710A, for example).
156 (341)	12, 13 (69, 70)	0FFC (17C7)	Hexadecimal representation of the rightmost four digits of the eight-digit controlware part number.
156 (341)	10, 11 (67, 68)	0FFB (17C6)	Hexadecimal representation of the leftmost four digits of the eight-digit controlware part number.
156 (341)	8, 9 (65, 66)	0FFA (17C5)	Controlware revision number.
156	6, 7	0FF9	Start of engineering change order (ECO)/field change order (FCO) list; each location contains the hexadecimal equivalent (for 844 controlware) or hexadecimal representation (for 667/669 controlware) of a decimal ECO/FCO number. The list begins with the most recent ECO/FCO number and ends with 0000 (for 844 controlware) or FFFF (for 667/669 controlware).
156	4, 5	0FF8	
156, 155	3 of 156, 77 of 155	0FF7	
155	75, 76	0FF6	
.	.	.	
.	.	.	
.	.	.	
155	13, 14	0FE1	
155	11, 12	0FE0	

SPECIAL INSTALLATION INFORMATION 5

This section gives special information for the installation and maintenance of the products. The products are listed in alphabetical order according to their installation procedure name. All products, except Cross, CCP, and the controlware, are listed, even if special information does not exist. When more than one product is installed by one installation procedure, the products are listed in alphabetical order under the common procedure name. Refer to table 2-1 for the names of the product installation procedures.

To get listings of the installation procedures, refer to Step 8 - Set Up Installation Files in section 2.

The installations of Cross, CCP, and the controlware differ from the installations of the other products. The installation of Cross and CCP is described in section 3. The installation of the controlware is described in section 4.

AAM2 — CYBER RECORD MANAGER ADVANCED ACCESS METHODS VERSION 2

When installing AAM2, code is assembled to gather additional file statistics if file USER includes the Update directive *DEFINE STATS. If this directive is omitted, the system gathers only normal file statistics.

To assemble code to interface with Sort/Merge 5 rather than Sort/Merge 4, specify SRT5 on the procedure call to AAM2.

HARDWARE REQUIREMENTS

Advanced Access Methods (AAM) requires 135000_8 words of central memory for installation.

ADDITIONAL PROCEDURES

AAM2 includes one system compression/decompression routine. You may add up to 53 additional compression/decompression routines as system routines. Encapsulate each added routine and modify the capsule OPNM$AA. The following procedure adds routines.

Each routine must have an entry point of the form CMPR$nn (nn is two decimal digits within the range 11 through 63). The first added routine's entry point name must be CMPR$11, the second routine's entry point name must be CMPR$12, and so forth. The entry point must be the second word (word 1) of the routine.

The first three words of each routine must have the format shown in table 5-1.

Table 5-1. Format of First Three Words of Compression/Decompression Routines

Word	Bits	Contents
0	59 through 18	Entry point name, 6-bit display code, left-justified, zero fill.
	17 through 0	1.
1	59 through 18	0.
	17 through 0	Starting address of compression code.
2	59 through 18	0.
	17 through 0	Starting address of decompression code.

An example of the construction of a single site-added compression/decompression routine follows:

```
         IDENT
         ENTRY   CMPR$11
         VFD     42/0LCMPR$11,18/1
CMPR$11  VFD     42/0,18/COMPRES
         VFD     42/0,18/EXPAND
         .
         .
         .
COMPRES  BSSZ       1
         .
         .
         .
         EQ      COMPRES
         .
         .
         .
EXPAND   BSSZ       1
         .
         .
         .
         EQ      EXPAND
         END
```

The CYBER Loader requires standard relocation for fast dynamic loading of capsules; therefore, construct the VFD statements as shown in the preceding example. A return jump to the address specified in word 1 or 2 of the routine effects the execution of the compression or decompression code.

For each added routine, add an entry to the capsule name table in deck OPNMDAA. The macro GENTBL (also part of OPNMDAA) generates the table entry and has the following format.

```
GENTBL name
```

Parameter	Description
name	Entry point name specified in word 0 of added routine.

Specify table entries in consecutive, ascending numerical order. For example, to add three routines, make the following change to OPNMDAA.

```
*B OPNMDAA.329
    GENTBL CMPR$11
    GENTBL CMPR$12
    GENTBL CMPR$13
*C OPNMDAA,DICODAA,CWEOR1,OPENDAA
```

To add one additional compression/decompression routine, execute a job similar to the following. Either it must be a system origin job, or you must have system origin privileges and have DEBUG on.

Job	Comments
job command.	
USER,username,password,familyname.	
CHARGE,*.	
LABEL,TAPE,F=I,VSN=REL3A.	
SKIPF,TAPE,44.	
COPYBF,TAPE,OLDPL.	
RETURN,TAPE.	
UPDATE,K.	
RFL,65000.	
COMPASS,A,I,S=TXTCRM,S=IPTEXT.	Assembles OPNMDAA and DICODAA.
SYMPL,ET=T,I,S=TXTCRM,S=IPTEXT.	Compiles OPNMDAA.
COMPASS,A.	Assembles routine being added.
RETURN,COMPILE.	
GROUP,$AAM$$CTL$. CAPSULE,$OPNM$$AA$. CAPSULE,$CMPR$$11$.	Encapsulates the modified capsule OPNM$$AA (deck OPNMDAA) and the new compression capsule.
LDSET,OMIT=$SETUP.$/$RM$$SYS=$.	
LOAD,LGO.	
NOGO,NEWCAP.	
COMMON,SYSTEM.	User must be validated to access common files.
GTR,SYSTEM,OLD.ULIB/AAMLIB	
LIBEDIT,B=0.	
LIBGEN,F=NEW,P=AAMLIB.	
SYSEDIT.	
--eor--	
*IDENT	
Update directives to modify OPNMDAA.	
*C OPNMDAA,DICODAA,CWEOR1,OPENDAA	
--eor--	
Compression/decompression routine being added (COMPASS).	
--eor--	
*DELETE CAP/OPNM$AA	
*FILE NEWCAP	
*BEFORE *,CAP/*	
--eor--	
*FILE AAMLIB	
--eoi--	

ALGOL5 — ALGOL-60 VERSION 5

No special information is needed to install ALGOL-60 5.

APL2 — APL VERSION 2

The installation of APL (A Programming Language) 2 consists of three procedures, APL2, APLUSRO, and APLUSR1. They must be run in the order listed.

The default character set for the system is the 64-character set. If a 63-character set is required, change IP.CSET in IPARAMS (refer to TEXT and TEXTIO in this section).

APL2 PROCEDURE

APL2 has IAF as its base system. The APL2 procedure may precede the APLUSRO and APLUSR1 procedures.

In the installation job for APL2, you may specify an additional parameter (TERMTYP=ttype) on the call to APL2. ttype defines the default terminal type and is one of the following.

The released default type is APLAS.

ttype	Description
APLAS	ASCII APL print (refer to APL 2 Reference Manual).
TYPEP	ASCII APL typewriter pairing.
BITPR	ASCII APL bit pairing.
ASCII	ASCII graphic 95-character set (not APL).
TTY33	Compatible with Teletype Model 33.
BATCH	Line printer with ASCII graphic 63/64-character set.
BH501	Line printer with CDC graphic character set.
TTY38	ASCII APL Teletype Model 38, model numbers 3841/4EA, 3841/4EG, 3851/6JA, and 3851/6JG.
CORRE	Compatible with IBM 2741.
CD713	ASCII 128-character set; lowercase alphabetic characters equate to uppercase alphabetic characters.

NOTE

Consult your local APL analyst for the most commonly used type of APL terminal at your site. Users can override the default terminal type by specifying the terminal type on the APL command.

To provide an APL entry message (a one-line message displayed when a user activates APL with a command), make a file named MESSAGE local just prior to the BEGIN,APL2,INSTALL...command. The file must contain only the one-line message, which must not exceed 80 characters in length.

APLUSR0 AND APLUSR1 PROCEDURES

Only the APL loader (on REL8B) can be captured on a deadstart tape. Except for the loader, APL 2 runs from a set of permanent files. Validate two user names, APL0 and APL1, to permit permanent file generation. Table 5-2 shows the recommended limits for the two user names.

Table 5-2. Recommended Limits for APL0 and APL1

Resource or Capability Mnemonic	User APL0		User APL1	
	Keyboard Entry	Converted Value	Keyboard Entry	Converted Value
MT	3	3	3	3
RP	2	2	2	2
TL	77B	Unlimited	77B	Unlimited
CM	40B	2037B	40B	2037B
DB	5B	10	5B	10
FC	7B	Unlimited	7B	Unlimited
CS	4B	4096	4B	4096
FS	6B	192	6B	192
PA	EVEN	Even	EVEN	Even
RO	37B	System	37B	System
PX	HALF	Half	HALF	Half
TT	TTY	TTY	TTY	TTY
TC	NORMAL	Normal	NORMAL	Normal
IS	NULL	Null	NULL	Null
MS	6B	25088	6B	25088
DF	73B	1008	73B	1008
CC	77B	Unlimited	77B	Unlimited
CP	77B	Unlimited	77B	Unlimited
LP	77B	Unlimited	77B	Unlimited
PT	77B	Unlimited	77B	Unlimited
EC	0B	0B	0B	0B
SL	77B	Unlimited	77B	Unlimited
CN				
PN				
DS	3B	1536	2B	1024
SC				
DT				
SP				
UP				
PW	APL0	APL0	APL1	APL1
AW	†	††	†	††

†Entry is CASF, CAND, CSRP, CPWC, CLPF, CSPF, CCNR
††Value is 000000000000000005~~40~~ 755

After validating user names APL0 and APL1, create workspace files for them with installation procedures APLUSR0 and APLUSR1. Execute procedure APLUSR0 before procedure APLUSR1. File USER must be local to both procedures and must contain USER and CHARGE commands.

Procedures APLUSR0 and APLUSR1 create SUBMIT jobs that require APL0 and APL1, respectively, as validated user names. If neither user name is valid, its respective job fails. The installation deck listing describes these jobs (refer to the second step in Step 8 - Set Up Installation Files in section 2).

BAM — CYBER RECORD MANAGER BASIC ACCESS METHODS VERSION 1

The following parameters are defined in the Update common decks /CMNTXT/ and /TXTCRM/. Assemble deck TXTCRM to obtain a listing of the common decks.

Parameter	Released Default Value	Significance
#LBLIM#	10D	Number of words in tape label buffer. Because each user label requires 9 words, set LBLIM to 9m+1; m is the maximum number of file header (HDRn) labels allowed. Minimum value is 10 words.
#CMU# #NOCMU#	1 1	Specifies use of compare and move unit (CMU) instructions in routine MOVE$RM. To remove the CMU code, delete the definition of CMU. To remove the no CMU code, delete the definition of NOCMU. If CMU and NOCMU are both defined, CYBER Record Manager determines at run time which MOVE routine to use by checking the CMU flag in RA.CMU. The use of CMU instructions reduces the execution time of a program using CYBER Record Manager for records of over 40 characters. The use of CMU instructions in programs to be executed on models 815, 825, 835, or 855 is not recommended.

BASIC3 — BASIC VERSION 3

The following parameter is defined in deck BASCOMP. Assemble this deck to obtain the Update sequence number required to change the released value.

Parameter	Released Default Value	Significance
BDFLT	1.0	Array base; can be any nonnegative value expressed as a real value.

The following parameters are defined in common deck LIPARAM. Assemble deck BASCARD to obtain a listing of LIPARAM.

Parameter	Released Default Value	Significance
MESSAG	1	Flag indicating whether BASIC issues time and memory use dayfile messages. A value of 0 inhibits issuing of messages; a value of 1 enables issuing of messages.
IP.AS	0	Flag indicating default character set mode. A value of 0 indicates normal (non-ASCII) mode (user must specify AS on the BASIC statement to override the default; a value of 1 indicates ASCII mode (user must specify AS=0 on the BASIC statement to override the default).
IP.BL	0	Flag indicating burstable listing. A value of 0 indicates nonburstable listing (user must specify BL on BASIC statement to override the default); a value of 1 indicates burstable listing (user must specify BL=0 on the BASIC statement to override the default).

Values specified in IPARAMS override the following parameters (also in LIPARAM). To obtain values differing from IPARAMS values for BASIC, supply the appropriate Update directives on file USER when installing BASIC to place local installation parameter definitions before the IPARAMS macro call in LIPARAM.

Parameter	Released Default Value	Significance
IP.PD	6	Print density in lines per inch. Value must be 6 or 8.
IP.PS	60	Page size in lines. Value must be greater than 3.

NOTE

The default settings for IP.PD and IP.PS work for both short (8 1/2 by 14 inch) and long (11 by 14 inch) paper. If you specify S (short) on the EST entry for the local batch printer (refer to Unit Record Equipment EST Entry in section 7), 60 lines per page will be printed; the print density parameter will be ignored. When using short paper, a page size specified at longer than 64 lines causes printing to continue over the page perforation. If users require 6-line per inch printing or a page size larger than the default, reserve one printer for long paper. Users can then route the print file with a special forms code to the correct printer.

BINEDIT — BINEDIT VERSION 1

No special information is needed to install BINEDIT.

BIT8 — 8-BIT SUBROUTINES VERSION 1

No special information is needed to install 8-Bit Subroutines.

CCG — COMMON CODE GENERATOR VERSION 1

No special information is needed to install Common Code Generator (CCG).

CCL — CYBER CONTROL LANGUAGE VERSION 1

CYBER Control Language (CCL) installation parameters and additional procedures follow.

INSTALLATION PARAMETERS

The following installation parameters are located on deck CCL. Because installation procedures use the default values, changing the values is not recommended.

Parameter	Released Default Value	Significance
IP.FPC	10	Maximum number of characters in a keyword for a procedure call or procedure header directive. Maximum value is 10.
IP.SCS	40	Maximum number of characters for default and actual values. Maximum value is 80.
IP.LCS	10	Maximum number of characters in a label string. Maximum value is 10.
IP.PNL	50	Procedure nesting limit. Maximum value is 1023.
IP.FP	50	Maximum number of keywords in a procedure call or procedure header directive. Maximum value is 500.
IP.DPF	1	Flag indicating logical existence of default procedure file name. Value — Definition 0 — No default procedure file name. 1 — Procedure file name defaults to value of IP.DPFN.
IP.DPFN	PROCFIL	Default procedure file name.

<table>
<tr><th>Parameter</th><th>Released Default Value</th><th>Significance</th></tr>
<tr><td>IP.TAPO</td><td>1</td><td>Flag indicating whether a procedure can reside on tape.
<table>
<tr><th>Value</th><th>Definition</th></tr>
<tr><td>0</td><td>Procedure file cannot reside on tape. BEGIN hangs in RECALL if execution from tape is attempted. A value of 0 decreases the execution size of CCL by 700_8 words for BEGIN, REVERT, WHILE, and ENDW.</td></tr>
<tr><td>1</td><td>Procedure file can reside on tape.</td></tr>
</table></td></tr>
<tr><td>IP.EXP</td><td>100</td><td>Number of operands and operators allowed in a CCL expression. For each unit that this parameter is decreased from 100, the execution size of CCL is reduced by two words.</td></tr>
<tr><td>IP.ATT</td><td>1</td><td>Flag indicating whether the system searches the user name's permanent file catalog, if the requested procedure file is not local.
<table>
<tr><th>Value</th><th>Definition</th></tr>
<tr><td>0</td><td>Permanent file catalog not searched.</td></tr>
<tr><td>1</td><td>Permanent file catalog searched. In order to attach the requested procedure file, the system searches the indirect access files first and then the direct access files.</td></tr>
</table></td></tr>
<tr><td>IP.NPV</td><td>6</td><td>Value used in the calculation of the size of the pattern value table (PVT). The PVT stores the checklist entries for each parameter in the procedure headers. The following formula determines the size of the PVT in words.

PVT = IP.NPV x IP.FP x 2</td></tr>
<tr><td>IP.RLD</td><td>1</td><td>Flag indicating whether the system does a sequential or random search of a library to find the requested procedure. A random search is usually faster than a sequential search.
<table>
<tr><th>Value</th><th>Definition</th></tr>
<tr><td>0</td><td>Search library sequentially.</td></tr>
<tr><td>1</td><td>Search library randomly by using the library directory.</td></tr>
</table></td></tr>
<tr><td>IP.SCL</td><td>150</td><td>Maximum length in characters of lines in a procedure. Any restrictions as to the length of a command remain in effect, but a comment following the command terminator may extend to the length specified by IP.SCL.</td></tr>
</table>

ADDITIONAL PROCEDURES

CCL consists of three absolute overlays with entry point names and verb table entries for each CCL verb (command). The CCL verbs and overlays are:

Overlay	Verbs
CCLBRWE	BEGIN, REVERT, WHILE, ENDW
CCLIFES	IFE, ELSE, ENDIF, SKIP
CCLDS	DISPLAY, SET

If a CCL verb must be changed due to a conflict with an existing program on the deadstart file, change both the entry point name and the verb table entry in the CCL overlay CCLBRWE.

CDCS2 — CYBER DATABASE CONTROL SYSTEM VERSION 2

CYBER Database Control System (CDCS) 2 operates at a system control point and it is initiated by operator DSD entries of the form:

```
ENABLE,CDC,n.
CDCffff.
```

n is the control point number at which CDCS 2 is to operate and ffff is from zero to four alphanumeric characters. The DSD entry executes a procedure whose file name must also be CDCffff. To automatically initiate CDCS 2 operations upon entry of the DSD AUTO command, name the procedure file CDCS and put it under the system user name SYSTEMX. Refer to the NOS 2 Operator/Analyst Handbook for further information on DSD commands; refer to section 9 for information on the IPRDECK entries that affect AUTO command processing.

CDCS 2 is structured with overlay capsules (OVCAPs) to reduce execution field length requirements when not all components are in use. In addition to a main overlay and 12 secondary overlays, there are 22 overlay-capsules for accounting, attach, automatic and basic recovery, quick recovery, journal logging, display/operator interface, privacy, and versions processing.

Procedure file CDCffff activates CDCS 2; it must be a public indirect access permanent file stored under the system user index (377777_8). Store the file under the system user index either by storing the file while running under the system user index at the system console (with the SUI command; refer to NOS 2 System Programmer's Instant) or by using the MOVEPF utility (refer to Step 13 - Move Files in section 2). Refer to the CYBER Database Control System 2 Reference Manual for instructions on constructing the procedure file. Include EXIT and DMD commands in the procedure file for PSR submittal and maintenance purposes, because CDCS 2 and its users operate at different control points.

CDCS 2 users must have permission to use the system control point facility (refer to the description of MODVAL in the NOS 2 System Maintenance Reference Manual).

Enable the system control point facility by entering ENABLE,SCP. in the IPRDECK (refer to section 9).

To activate a debug trace facility for CDCS 2, specify the E parameter on the SYMPL commands in the CDCS2 installation procedure. To activate commands that generate CDCS flow points, specify the keyword DEBUG on the call to CDCS2. These flow points trace the execution of CDCS modules from initialization to termination. Generation of the flow points increases the execution size of CDCS by approximately 2500_8 words.

To assemble code to interface with Sort/Merge 5 rather than Sort/Merge 4, specify the keyword SRT5 on the call to CDCS2. User library SRT5LIB must be available during execution of the installation procedure.

For more information on activating the interface between CDCS 2 and COBOL5, refer to COBOL5 in this section.

ACCOUNTING TABLE

The CDCS routine DB$ACCT contains a table of average central processor (CP) and input/output (I/O) times, in microseconds, for CDCS user requests. These average values were obtained from simulation runs on a model 74 and adjusted based on actual runs performing file creation and updating on indexed sequential files with a record size of 40 words.

When a user issues a CDCS request, such as open, read, or rewrite, CDCS retrieves the value from the appropriate table entry and accumulates it in the accounting accumulator for the individual user. CDCS accumulates the charged CP and I/O time for all users combined and prints it in the dayfile at the end of the CDCS session. The actual time used for the entire CDCS session is also printed in the dayfile. Because different environments produce different values for the average CP and I/O times required for each user request, CDCS provides options for the data base administrator to modify these table values.

One method of modification is specifying new values for the CP and I/O times on the command that initializes CDCS (refer to the CDCS 2 Data Administration Reference Manual). With this method, all the entries in the accounting table are multiplied by the ratio of the specified value to the table value for a random read on an indeed sequential file.

A second method of modification is changing the values in the accounting table and installing CDCS with the recompiled table. You can modify any or all entries in the table. List the deck DB$ACCT to see the current values in the accounting table. Entries in the table are in COMPASS macro format, as follows.

Field 1 (column 1): Blank or, if continuation of the previous line, a comma.

Field 2 (beginning in column 2): One of the following user request codes.

DFLOG	Logging
DFRD2	Random Read
DFRD1	Sequential Read
DFWR2	Random Write
DFSKF	Skip
DFREW	Rewrite
DFDEL	Delete
DFOPN	Open
DFCLS	Close
DFSTX	Start on index file

DFINV	Invoke
DFSTR	Start
DFEND	End
DFTER	Abnormal termination
DFRPT	Recover point
DFPVC	Privacy
DFLOK	Lock
DFULK	Unlock
DFRSR	Relation start
DFDBS	Database status block
DFRX2	Read random on index file
DFRX1	Read sequential on index file
DFRWX	Rewind index file
DFRWF	Rewind area file
DFRWR	Rewind relation
DFVER	Version change
DFBEG	Begin transaction
DFCMT	Commit transaction
DFDRP	Drop transaction
DFASK	Ask restart identifier
DFGID	Get restart identifier

Field 3 (beginning in column 11): The macro identifier ACC

Field 4 (beginning in column 18): CP and I/O times required by each request. Parameters represent the different types of charges according to different file organizations, logging, and other factors. Possible parameters are as follows.

AK	AK primary key charge
ALT	Alternate key charge
ARL	Area logging flag
DA	DA primary key charge
FIX	Fix charges

IS	IS primary key charge
JLG	Journal logging charge
JNL	Journal logging flag
MOD	Database modification flag
QLG	Quick recovery logging charge
QRF	Quick recovery logging flag

The following is an example of an entry in the table.

```
DFRD2  ACC  ((IS=4000,7000),(DA=3500,6500),(AK=3000,6000),(ALT=3000,7000))
```

This entry states that for a random read performed on (for example) an indexed sequential file, the CP charge will be 4000 microseconds, and the I/O charge will be 7000 microseconds.

VERIFICATION

To verify the installation of CDCS 2, do the following:

1. Run a job that executes the following command.

   BEGIN,VCDCS2A,INSTALL,S=username.

Parameter	Description
username	User name used for installation work.

   This job creates the permanent files SCIO, MSTRDIR, and CDCS2 required for CDCS 2 operation during verification.

2. Enter the following DSD commands at the system console.

   X.MOVEPF(F1=CDCS2/UI=ui,DI=377777)
ENABLE,CDC,n.
CDCffff.

Parameter	Description
ui	User index corresponding to user name specified in step 1.
n	Control point.

   ffff is from zero through four alphanumeric characters.

3. Run a job that executes the following command.

   BEGIN,VCDCS2B,INSTALL.

   This job creates the permanent file IOAREAB. If this job is successful, CDCS 2 verification is complete.

4. Access the K display with the following DSD command.

   K,CDC.

5. Enter the following command to terminate CDCS2.

   K.TERM.

CEDIAG — CEDIAG VERSION 1

No special information is needed to install CEDIAG 1.

CID — CYBER INTERACTIVE DEBUG VERSION 1

The following parameters define the size of various tables used by CYBER Interactive Debug (CID). Certain table sizes are defined by parameters in both SYMPL and COMPASS decks. If you alter such a table size, change all installation parameters defining the table size. Compile or assemble the indicated Update deck(s) to obtain sequence information.

Parameter	Released Default Value	Significance
BREAKTABSIZE TABSIZE	16 16	Number of entries in breakpoint table. Parameters are located in common decks BREAKD (SYMPL) and BREAKZ (COMPASS).
GROUPTABSIZE TABSIZE	16 16	Number of entries in group table. Parameters are located in common decks GROUPD (SYMPL) and GROUPZ (COMPASS).
TRAPTABSIZE TRAPXSIZE TABSIZE XSIZE	16 19 16 19	Number of entries in trap table. TRAPX-SIZE and XSIZE must each be three greater than the table size defined by TRAPTABSIZE and TABSIZE. Parameters are located in common decks TRAPD (SYMPL) and TRAPZ (COMPASS).
ROOM54	10B	Number of words available for EACPM loader table (54 table) expansion before CID must recreate its overlays at debug time. Parameter is located in deck DBUGI.

COBOL5 AND COBOL5Q — COBOL VERSION 5 SEE ERRATA 5-73

COBOL Version 5 has two methods of installation: the full mode installation, which assembles and compiles all compiler and object library routines, and the Q mode installation (deck COBOL5Q).

Q mode installation allows the user to modify only those routines affected by corrective code and/or user code (for example, activating Data Management) and local code (for example, default page size, CMU, and so forth), and to produce a new file through the COPYL utility, using the changed routines and a previous release level of COBOL 5 as input. The user is responsible for providing *COMPILE directives on file USER for any affected routine.

You should examine installation procedure COBOL5Q before using it.

INSTALLATION PARAMETERS

The COBOL 5 compiler uses IPTEXT symbol definitions, which are filtered through CB5TEXT. No direct references to any IPTEXT symbols are contained in the compiler or the object routines. This allows you more flexibility in changing normal installation parameters for COBOL 5.

The system obtains symbols governing machine type, character set, and CMU option from IPTEXT. You can override the system defaults for activating CDCS processing, default page size, print density, and default error termination for COBOL 5. To override one or more of these system defaults, select the desired changes from the following list and put them on file USER for the COBOL5 or COBOL5Q installation procedure.

- To create a compiler that generates code for a machine with CMU, insert the following code after the *OPTION= and before the OP.BDP label in deck CB5TEXT.

  ```
  OP.BDP    EQU    OP.YES
  ```

- To create a compiler that generates code for a machine without CMU, insert the following code after the *OPTION= and before the OP.BDP label in deck CB5TEXT.

  ```
  OP.BDP    EQU    OP.NO
  ```

- To change the default error termination level to T, W, F, or C, use 1, 2, 3, or 4, respectively, for level in the following statement. The DEF CB5$ET statement is in deck ASSEMOP.

  ```
  DEF CB5$ET#level#;
  ```

- To activate CDCS processing, perform the following two steps.

  1. Include a *PURGE DMGMNT directive and change the statement with label OP.DCS in deck CB5TEXT to:

     ```
     OP.DCS    EQU    OP.DCS2
     ```

  2. Change the DEF CB5$CDCS statement in deck ASSEMOP to:

     ```
     DEF CB5$CDCS #≠CDCSn≠#;
     ```

  Results are unpredictable if both changes are not made.

  If you install COBOL 5 using the released program library on REL5C, no CDCS processing is activated. If you install COBOL 5 from the released binary, CDCS 2 processing is activated.

- Print density is determined by one of the following factors, in descending order of dominance. The dominant factor is the PD parameter on the COBOL5 command; next is the installation-specified value of CB5$PDENS (if other than zero); and last is the value of IP.PD in IPTEXT.

To select a default print density different from that specified in IPTEXT, find the line for CB5$PDENS in deck ASSEMOP and make the following change.

DEF CB5$PDENS #n#;

Parameter	Description
n	Lines per inch; n can be 3, 4, 6, or 8.

- The number of lines per page is determined by one of the following factors, in descending order of dominance. The dominant factor is the PS parameter on the COBOL5 command; next is the installation-specified value of CB5$LINP (if other than zero); and last is the result of the following calculation.

  Lines per page = print density x (IP.PS/IP.PD)

  Print density is the density determined from the previously described factors. IP.PS and IP.PD are the values defined in IPTEXT.

  It is not necessary to change CB5$LINP. However, to do so, locate CB5$LINP in deck ASSEMOP and change it to:

  DEF CB5$LINP #n#;

Parameter	Description
n	An integer. The page will contain n lines, including three lines at the top and three lines at the bottom for headings.

NOTE

The default settings for IP.PD and IP.PS work for both short (8 1/2 by 14 inch) and long (11 by 14 inch) paper. If you specify S (short) on the EST entry for the local batch printer (refer to Unit Record Equipment EST Entry in section 7), 60 lines per page will be printed; the print density parameter will be ignored. When using short paper, a page size specified at longer than 64 lines causes printing to continue over the page perforation. If users require 6-line per inch printing or a page size larger than the default, reserve one printer for long paper. Users can then route the print file with a special forms code to the correct printer.

- To change the CPU type for which code is generated and object routines are assembled, insert the following statement after the *OPTION= statement and before the OP.MODEL label in deck CB5TEXT.

```
OP.MODEL  EQU   OP.machine
```

Parameter	Description
machine	6400 for a computer with a unified CPU; 6600 for a computer with a nonunified CPU.

- To change the default organization (xx) for actual key, direct access, or indexed (IS) files from version 2 (ORG=NEW) to version 1 (ORG=OLD), locate CB5$xxOLDNEW in deck ASSEMOP and change it to:

```
DEF CB5$xxOLDNEW #≠OLD≠#;
```

xx	Description
AK	Actual key files.
DA	Direct access files.
IS	Indexed files.

ADDITIONAL PROCEDURE

To activate the interface between COBOL 5 and TAF, specify TAF on the call which executes the installation procedure for COBOL 5.

COMPASS — COMPASS VERSION 3

The common common decks are on the COMPASS program library; the COMPASS installation procedure places them on permanent file COMCPL. Control access to COMCPL as required by your site's license agreement.

You can set the following parameters either in IPTEXT or at CMP30.62 in the update of REL3A.

Parameter	Released Default Value	Significance
IP.PD	6	Print density in lines per inch. COMPASS supports either six or eight lines per inch.
IP.PS	60D	Specifies lines per page for COMPASS listable output. COMPASS supports from 4 through 99 lines per page.

NOTE

The default settings for IP.PD and IP.PS work for both short (8 1/2 by 14 inch) and long (11 by 14 inch) paper. If you specify S (short) on the EST entry for the local batch printer (refer to Unit Record Equipment EST Entry in section 7), 60 lines per page will be printed; the print density parameter will be ignored. When using short paper, a page size specified at longer than 64 lines causes printing to continue over the page perforation. If users require 6 line per inch printing or a page size larger than the default, reserve one printer for long paper. Users can then route the print file with a special forms code to the correct printer.

DBU — DATABASE UTILITIES VERSION 1

No special information is needed to install Database Utilities.

DCAT2 — DATA CATALOGUE VERSION 2

The COBOL 5 compiler and library must be available for the installation of Data Catalogue 2. The product must run from permanent files.

The installation procedure (DCAT2) uses the released Data Catalogue 2 tape to create an output tape containing all the modules necessary to execute Data Catalogue 2. To make Data Catalogue files available to users, run a job similar to the following after execution of the DCAT2 installation procedure.

Job	Comment
job command.	
USER,username,password,familyname.	
CHARGE,*.	
LABEL,TAPE,D=xx,yy,LB=KL,VSN=zz.	xx is the tape density, yy is MT or NT (seven- or nine-track respectively),and zz is the tape VSN.
SKIPF,TAPE,2.	
PURGE,DCUPD,DCSEL,DCRET,DCCONVT/NA.	
PURGE,DCRPT,DCUTL,DCIDX,DCCONGN/NA.	
PURGE,DCGEN/NA.	
DEFINE,DCUPD,DCSEL,DCRET,DCCONVT/CT=PU,M=R.	
DEFINE,DCRPT,DCUTL,DCIDX,DCCONGN/CT=PU,M=R.	
DEFINE,DCGEN/CT=PU,M=R.	
COPYBF,TAPE,DCUPD.	
COPYBF,TAPE,DCSEL.	
COPYBF,TAPE,DCRPT.	
COPYBF,TAPE,DCRET.	
COPYBF,TAPE,DCCONVT.	
COPYBF,TAPE,DCUTL.	
COPYBF,TAPE,DCIDX.	
COPYBF,TAPE,DCGEN.	
COPYBF,TAPE,DCCONGN.	
UNLOAD,TAPE.	
--eoi--	

DDL3 — DATA DESCRIPTION LANGUAGE VERSION 3

No special information is needed to install Data Description Language (DDL) 3.

FCL1, FCL2, AND PMD4 — FORTRAN COMMON LIBRARY VERSION 4 WITH POSTMORTEM DUMP UTILITY

FORTRAN Common Library (FCL) 4 includes mathematical, input/output, postmortem dump (PMD), and miscellaneous routines that are used by FORTRAN Extended 4, PL/I, SYMPL, COBOL 5, and Data Description Language 3.

FORTRAN Common Library 4 uses the definition of IP.CSET from IPARAMS. No other installation-changeable parameters exist.

Because FORTRAN Common Library 4 is incrementally installed by procedures FCL1 and FCL2, include any site code in both procedures.

No special information is needed to install PMD4.

FCL5 AND PMD5 — FORTRAN COMMON LIBRARY VERSION 5 WITH POSTMORTEM DUMP UTILITY

FORTRAN Common Library 5 includes mathematical, input/output, character, PMD, and miscellaneous routines that are used by FORTRAN 5.

FORTRAN Common Library 5 uses the definition of IP.CSET from IPARAMS. No other installation-changeable parameters exist.

No special information is needed to install PMD5.

FCS3 — CONVERSION AIDS SYSTEM VERSION 3

Refer to LCS3 in this section for the product modification information.

FDBF — FORTRAN DATA BASE FACILITY VERSION 1

The installation tool SYNGEN, which resides on the DDL 3 program library, must be available for the installation of FORTRAN Data Base Facility (FDBF) 1.

FDBF 1 supports either FORTRAN Extended 4 or FORTRAN 5. The user specifies the FORTRAN version on the DDLF and DML commands. If the language version is not specified on those commands, the released default is FORTRAN Extended 4. To change the default to FORTRAN 5, add the following directives.

```
*D P132.13†
    DDLCOMP EPT     F5††
*D P180.9
    DDLCOMP=F5; #FIRST DEFAULT FORTRAN VERSION (LV)#
*D P180.10†
    DDLCOMP=F4; #SET COMPILATION LANGUAGE MODE TO FORTRAN EXTENDED 4#
```

†P132.13 is located after ARG.131; P180.9 is located after DML.151; P180.10 is located after DML.230.

††F5 specifies the compilation language mode.

This code causes the following changes in command processing:

- For the DDLF command, if the user specifies neither F4 nor F5, the default value is F5.
- For the DML command, if the user omits the LV= parameter, the default value is F5. If the user specifies only LV, the default is F4.

FORM — FORM VERSION 1

No special information is needed to install File Organizer and Record Manager (FORM) 1.

FORMAT — 881/883 PACK FORMATTING

No special information is needed to install 881/883 Pack Formatting.

FTN AND FTNTS — FORTRAN EXTENDED VERSION 4 AND FORTRAN EXTENDED VERSION 4 WITH INTERACTIVE OPTION

FORTRAN Extended 4 with Interactive Option is released separately on REL4B. It contains an additional interactive mode (TS). When the user selects TS on the FTN command, FTN operates as a one-pass compiler instead of a two-pass compiler. The following discussion applies to both FORTRAN Extended 4 and FORTRAN Extended 4 with Interactive Option.

Both FORTRAN Extended 4 and FORTRAN Extended 4 with Interactive Option reference the MODEL parameter (refer to TEXT and TEXTIO procedures in this section). Whether a computer will efficiently execute the FORTRAN object code that it produced depends upon the model of the computer and the value specified in the MODEL parameter. If the value specified in the MODEL parameter is identical to the computer's model number, the object code executes efficiently. If the value specified in the MODEL parameter is different from the computer's model number, the object code executes inefficiently or not at all, as table 5-3 shows.

Table 5-3. MODEL Parameter and FTN, FTNTS, and FTN5 Object Code Execution

Value of MODEL Parameter in TEXT and TEXTIO	Object Code Executes Inefficiently on These Models	Object Code Does Not Execute on These Models
71, 72, 73, 171, 172, 173, 174, 720, 730	74, 175, 740, 750, 760, 815, 825, 835, 855, 865, 875	176
74, 175	71, 72, 73, 171, 172, 173, 174, 720, 730, 740, 750, 760, 815, 825, 835, 855, 865, 875	176
176	All models, except model 176, as long as source code does not contain LEVEL 2 (direct access LCM) statements.	All models, except model 176, if the source code contains LEVEL 2 statements.

Depending on the installation parameters of interest, you can obtain the parameters by assembling FTNMAC or FTNTEXT (the FTNMAC listing is much shorter) and/or FTN. FTN contains the installation parameters for default command settings, command error processing, default file names, input/output buffer length, overlay library names, and reduce mode field length increments. The remaining parameters are in OPTIONS (called by FTNMAC/FTNTEXT).

FTN5 — FORTRAN VERSION 5

Because all code generated by the compiler assumes the existence of the integer multiply hardware option, you must install all applicable integer multiply FCOs.

FORTRAN 5 references the MODEL parameter (refer to TEXT and TEXTIO in this section). Whether a computer will efficiently execute the FORTRAN object code that it produced depends upon the model of the computer and the value specified in the MODEL parameter. If the value specified in the MODEL parameter is identical to the computer's model number, the object code executes efficiently. If the value specified in the MODEL parameter is different from the computer's model number, the object code executes inefficiently or not at all, as table 5-3 shows.

Most user programs written in FORTRAN Extended 4 require translation before they compile properly under FORTRAN 5. Refer to the FORTRAN Extended 4 to FORTRAN 5 Conversion Aids Program Reference Manual for a product description.

Depending on the installation parameters of interest, you can obtain the parameters by assembling FTN5TXT and/or FTN. FTN contains the installation parameters for default command settings, command error processing, default file names, input/output buffer length, and compiler overlay library names. The remaining parameters are in OPTIONS (called by FTN5TXT).

Reinstall the compiler and CCG whenever you change parameters in OPTIONS. You can revise installation parameters in COMFCIP (called by FTN and INIT00) during the installation of FORTRAN 5, if you reassemble both FTN and INIT00.

F45 — FORTRAN 4 TO 5 CONVERSION AID VERSION 1

No special information is needed to install FORTRAN 4 to 5 Conversion Aid. This conversion aid is a translator program, which assists the user in converting FORTRAN Extended 4 source programs to FORTRAN 5 source programs.

NOTE

Input to the FORTRAN 4 to 5 Conversion Aid must be valid, diagnostic-free, FORTRAN Extended 4 source programs that comply with the specifications given in the FORTRAN Extended 4 Reference Manual.

HSIO - HIGH SPEED I/O OPTION

No special information is needed to install the High Speed I/O option.

IAF - INTERACTIVE FACILITY VERSION 1

Information on installation parameters, file placement, and the GENHELP procedure follows.

INSTALLATION PARAMETERS

The following parameters, defined in deck IAFEX, specify default values for the Application Interface Program (AIP) Trace utility in IAF.

Parameter	Released Default Value	Significance
DMCT	16200	Maximum number of messages logged before the trace file is released to the system for processing.
MXML	10	Maximum length in central memory words of a message logged on the trace file.
TJOB	TRACIAF	Micro whose string specifies the name of the procedure file containing the job commands used to process the trace file.

When you call the installation procedure IAF, the procedure assembles IAF and moves the system procedures IAF, IAFTM, and IAFTR from RELSOPL to the file PRODUCT and subsequently to the deadstart tape. When IAFTM or IAFTR are used, they store a copy of TRACIAF under the system user index (377777_8) for subsequent use.

NOTE

The OPL deckname for the IAF procedure is IAFP.

Procedure file LTRACIAF contains the commands to process the trace file.

Modify procedure TRACIAF so that the USER and CHARGE commands have the appropriate parameters. The modifications can be placed on file USER.

Two additional procedures exist to enable the console operator to select the type of trace, according to the parameter specified on the IAFEX command. In procedure IAFTM, T=* is the parameter on the IAFEX command; it causes the trace file to be processed only when IAF has terminated. In procedure IAFTR, T is the parameter on the IAFEX comand. The T parameter causes the trace file to be submitted as an input job using the TRACIAF file for the command record after every 16200 messages hae been logged on the trace file. Refer to NOS 2 System Maintenance Reference Manual for a description of the IAFEX command.

To initiate IAF operations, the operator enters the DSD command:

```
ENABLE,IAF.
IAFffff.
```

The name of the IAF procedure file and the DSD command to initiate IAF must be the same and must be of the form IAFffff (ffff is from zero through four alphanumeric characters). If the system is to initiate IAF operations automatically when the operator enters the DSD command AUTO, the procedure file must have the three-character name IAF, and you must have enabled IAF in the IPRDECK.

You can create procedure files different from those created by the installation procedure and select them with different operator commands. For additional information regarding the trace utility, refer to the NOS 2 System Maintenance Reference Manual.

GENHELP PROCEDURE

GENHELP is the procedure on the system that generates the file of command explanations for the HELP program that runs under IAF. GENHELP creates the file and stores it as a permanent file under the user name LIBRARY,. To initiate GENHELP, enter the following at the system console.

```
X.DIS
GENHELP,#D=density.
```

IMF1 — INFORMATION MANAGEMENT FACILITY VERSION 1

The installation tool SYNGEN, which resides on the DDL 3 program library, must be available for the installation of Information Management Facility 1 (IMF1).

To assemble code to interface with Sort/Merge 5 rather than Sort/Merge 4, specify SRT5 on the call to IMF1.

LCS3 AND FCS3 — CONVERSION AIDS SYSTEM VERSION 3

The Conversion Aids System (CAS) includes the Language Conversion Aids System (LCS) and the File Conversion Aids System (FCS).

HARDWARE REQUIREMENTS

CAS can be maintained on a 49K hardware configuration for NOS. If you choose installation parameters other than the default parameters, more central memory is required.

INSTALLATION PARAMETERS FOR LCS3

The tables of the FORTRAN and COBOL language conversion processors (LCPs) may overflow when programs with large numbers of symbols or with lengthy statements are processed.

The FORTRAN LCP name table contains a fixed-size entry for each name that appears in a declarative statement. The COBOL LCP name table contains a variable-size entry for each special name, file name, and data name, except within either an RD entry in the Report Section or an SD entry in the File Section. COBOL name table entries are 4+(n+9)/10 words long (n is the number of characters in the name), with no rounding up. For example, if n=4, the name table entry is 5 words; if n=20, the name table entry is 6 words.

You can enlarge these tables by including either of the following Update directives on file USER in the installation job for LCS.

```
*DEFINE LTAB
*DEFINE LTAB,XLTAB
```

Table sizes and central memory requirements are shown in table 5-4.

Table 5-4. Table Sizes and Central Memory Requirements

LCP Name Table	No *DEFINE (Default)	*DEFINE LTAB	*DEFINE LTAB,XLTAB
FORTRAN			
Table size	300 entries	600 entries	-
Minimum central memory required	61000_8 words	65000_8 words	-
COBOL			
Table size	3200 words	7000 words	13000 words
Minimum central memory required	60000_8 words	70000_8 words	106000_8 words

To create a special version of the LCS that includes the COPY processing logic and additional CRM routines, make the following Update directive available on file USER when the job to install the LCS is run.

```
*DEFINE CBLCOPY
```

The central memory requirements for this version of the LCS are increased by approximately 20400_8 words.

To create a special version of the LCS that generates COBOL sequence numbers in increments of 1 (the default is 10), make the following Update directive available on file USER when the job to install the LCS is run.

```
*DEFINE COLUMN6
```

The central memory requirements for this version are increased by five words.

INSTALLATION PARAMETERS FOR FCS3

No special information is needed to install File Conversion Aids System.

LOADER — CYBER LOADER VERSION 1

You can change the following parameters for CYBER Loader. Insert the parameter changes at LDRCOM.13 in the update of REL3A.

Parameter	Released Default Value	Significance
IP.PSET	11B	Central memory presetting options; one of the following values.

Value	Preset (Octal)
0	No presetting for ECS; same as 1 for CM.
1	0000 0000 0000 0000 0000
2	7777 7777 7777 7777 7777
3	1777 0000 0000 0000 0000
4	3777 0000 0000 0000 0000
5	6000 0000 0000 0000 0000
6	4000 0000 0000 00xx xxxx †
7	2525 2525 2525 2525 2525
10	5252 5252 5252 5252 5252
11	6000 0000 0004 00yy yyyy ††

If you install CYBER Loader using the program library on REL3A, the value of IP.PSET is 11_8. If you install the binary on REL3A, the value is 1 (preset to all zeros).

Parameter	Released Default Value	Significance
IP.REW	1	Specifies whether file is rewound prior to beginning of load; one of the following values.

Value	Description
1	File is rewound.
0	File is not rewound.

Parameter	Released Default Value	Significance
IP.LDER	1	Error processing by the loader; one of the following values.

Value	Description
0	Abort on all errors (ERR=ALL).
1	Abort on fatal errors (ERR=FATAL).
2	No abort if abort is possible (ERR=NONE).

Parameter	Released Default Value	Significance
IP.FLINC	4000B	Amount of field length increase if additional field length is required for table construction by the loader. Acceptable values are multiples of 100_8.

†xx xxxx is the address of the preset location.
††yy yyyy is the result of adding 400000_8 and the address of the preset location.

Parameter	Released Default Value	Significance
IP.LRT	0	If nonzero, a message giving various time and memory measurements is issued to the dayfile.
IP.LDBG	0	If nonzero, conditional code to aid in debugging the loader is assembled.
IP.FLMSG	0	If nonzero, a dayfile message giving field length required for loading and execution is issued; applies for relocatable loads when no map is specified.
IP.MAP	3	Default loader map option; one of the following values.

Value	Description
0	Specifies MAP(OFF): no map.
3	Specifies MAP(PART): statistics and block map.
13B	Specifies MAP(ON): statistics, block map, and entry point cross-references.
17B	Specifies MAP(FULL): statistics, block map, entry point map, and entry point cross-references.

If you install CYBER Loader using the program library released on REL3A, the default value is 3. However, if you install CYBER Loader using the binary on REL3A, the default value is 0.

CYBER Loader also uses the symbols IP.PD, IP.PS, and IP.MECS, which are defined in IPARAMS during the installation of TEXT and TEXTIO.

MCS — MESSAGE CONTROL SYSTEM VERSION 1

A procedure file MCSffff must be created to execute MCS. This file is executed when the operator enters the DSD commands

```
ENABLE,MCS,n.
MCSffff.
```

to initiate Message Control System (MCS) operations. n is the control point number at which MCS is to execute. The name of the procedure file and the DSD command to initiate MCS processing must be the same and of the form MCSffff (ffff is from zero through four alphanumeric characters). If the system is to initiate MCS operations automatically when the operator enters the DSD command AUTO, the procedure file must have the three-character name MCS, and MCS must be enabled in the IPRDECK. MCSffff must be either an indirect access permanent file stored under the system user index (377777_8) or a system resident procedure. Use the MOVEPF utility to put the procedure file under the system user index (refer to Step 13 - Move Files in section 2).

Parameters in the procedure control the following aspects of MCS initialization.

- Default user name and family.
- Default Application Definition Language (ADL) file name.
- Operator interaction.

The default user name for MCS is SYSTEMX. To change the user name, insert a USER command in the procedure that specifies the user name and family under which MCS is to run.

To change the default ADL file, include an ATTACH or GET command in the procedure so that the local file name ADLLIB exists before MCS is called. If file ADLLIB does not exist locally, MCS tries to acquire a file with the name ADLLIB under either the default user name or the name specified with the USER command.

Inclusion of a GO parameter on the MCS program call command prevents operator interaction during MCS initialization.

Consider the following two procedures.

Example 1:

```
.PROC,MCS.
RETURN,MCS.
RFL,60000.
MCS,GO.
EXIT.
REWIND,ZZZZZDN.
DLFP,I=0.
```

Example 2:

```
.PROC,MCSTEST.
USER,username,password,familyname.
CHARGE,*.
RFL,60000.
ATTACH,ADLLIB/UN=username.
MCS.
EXIT.
REWIND,ZZZZZDN.
DLFP,I=0.
```

The procedure named MCS specifies the default user name (SYSTEMX) and the default ADL file (ADLLIB); it does not allow the operator to change initialization parameters.

The procedure named MCSTEST specifies a different user name, family name, and ADL file and allows the operator to change initialization parameters. The call to DLFP is required only if you use a debug version of MCS.

SPECIAL NOTES

When installing MCS, specify the keyword TRACE on the call to MCS to activate a debug trace facility. Use this only if you are thoroughly familiar with MCS.

To activate debug dumps for the Application Definition Language processor, include a *DEFINE DEBUG directive on file USER for the MCS installation job.

Users must be validated to access MCS (refer to MODVAL in the NOS 2 System Maintenance Reference Manual).

INSTALLATION PARAMETERS

The following parameters are defined in common deck IPA$MCS. To change these parameters, place the appropriate Update directives on file USER for the MCS installation job.

Parameter	Released Default Value	Significance
MAXFL	110000	Maximum field length (octal) to which MCS can expand.
OUTLIMIT	60	Number of messages that can accumulate in an output queue before SEND requests are rejected.

The following parameters assign relative weights to the various requests that a COBOL program can make to MCS. When the program disconnects from MCS, the accounting routine adds the corresponding weight factors of all requests and enters the total into the system account file.

Parameter	Released Default Value	COBOL Request
AC$ACCEPT	1	Accept.
AC$CHECKPT	1	Check point.
AC$DISABLE	1	Disable.
AC$ENABLE	1	Enable.
AC$INITIAL	2	Initial.
AC$PURGE	2	Purge.
AC$RECEIVE	3	Receive.
AC$SEND	3	Send.
AC$STOPRUN	2	Stop run.

VERIFICATION

The following procedure verifies the correct installation of MCS, the ADL processor, and the COBOL communications facility.

Run a job that executes the following command.

```
BEGIN,VMCS1A,INSTALL.
```

This job creates an application definition library file (ADLLIB). Use the MOVEPF utility (refer to Step 13 - Move Files in section 2) to place ADLLIB under the system user index (377777_8). For the UI=ui parameter, use either the user index under which VMCS1A runs or the user name that is specified for ADLLIB when an ATTACH or GET command is included in the initialization procedure.

To start MCS processing, enter:

```
ENABLE,MCS,n.
MCSffff.
```

n is the control point number, and ffff is from zero through four alphanumeric characters.

Then, to compile a sample COBOL job and execute MCS-related verbs, run a job that executes the following command.

```
BEGIN,VMCS1B,INSTALL.
```

Optionally, you can perform a part of the verification procedure at a terminal. Bring up NAM by entering at the system console the following two commands.

```
ENABLE,NAM,n.
NAMffff.
```

n is the control point number, and ffff is from zero through four alphanumeric characters. When NAM is up, log in at a terminal and specify MCS as the network application. After the MCS banner and prompt appear, enter

```
VERIFY AOP
```

as the MCS application name and

```
TERMINAL1
```

as the symbolic name. A verification message appears at the terminal.

After verification is complete, enter

```
CFO,MCS.IDLE
```

to terminate MCS. To terminate NAM, assign the NAM K display to NVF and enter the following K display command.

```
DI,HO
```

MMF - MULTIMAINFRAME MODULE VERSION 1

Refer to the NOS 2 System Maintenance Reference Manual for information concerning system operation in a multimainframe environment.

No special information is needed to install the Multimainframe Module (MMF).

MSS - MASS STORAGE SUBSYSTEM VERSION 1

When you call the installation procedure MSS, the procedure assembles the product and moves the system procedure MSS from RELSOPL to the file PRODUCT. The keyword SAVELIB on the call to install MSS saves MSSLIB as a direct access file. To initiate MSS operations, the operator enters the DSD commands:

```
ENABLE,MSS,n.
MSSffff.
```

n is the control point number at which MSS is to run. The name of the MSS procedure file and the DSD command to initiate MSS processing must be the same and must be of the form MSSffff (ffff is from zero through four alphanumeric characters). If the system is to initiate MSS operations automatically when the operator enters the DSD command AUTO, the procedure file must have the three-character name MSS, and MSS must be enabled in the IPRDECK. MSSffff must be either an indirect access permanent file stored under the system user index (377777_8) or a system resident procedure.

NAM5 - NETWORK ACCESS METHOD VERSION 1

NAM operations are enabled by the following two DSD entries.

```
ENABLE,NAM,n.
NAMffff.
```

n is the control point number, and ffff is from zero through four alphanumeric characters. NAMffff is a procedure stored in an indirect access permanent file under the system user index (377777_8) or a system resident procedure. If NAM operations are to be initiated automatically upon entry of the DSD AUTO command, name the file NAM, and have an ENABLE,NAM,n. entry (n is the control point number) in the IPRDECK.

NAMffff causes the network initialization program NAMI to execute. NAMI takes input from a permanent file which it creates the first time it executes (under user name SYSTEMX), from its command parameters, and from the operator through CFO commands. Command parameters override the permanent file; operator entries override both the command parameters and the permanent file.

The parameters to the NAMI command primarily identify a startup master file that NAMI uses to bring up network host programs. A GO parameter on the NAMI command brings up the network without operator intervention. This parameter is provided on the NAMI command in the NAM procedure, but it is absent in the NAMNOGO procedure.

The first time the network programs are to be started, the operator should enter NAMNOGO to allow entry of the parameters necessary for it to execute. These parameters are the name of the startup master file, the user name and password under which the file resides, and the name of the parameter record on the master file containing additional directives to NAMI for starting the network programs. The operator enters the required parameters through the CFO command, ending with the GO parameter to start NAMI processing. Refer to the NOS 2 System Maintenance Reference Manual for a description of the NAMI command and to the NOS 2 Operator/Analyst Handbook for further details on NAMNOGO and the CFO command.

The released startup master file, NAMSTRT, is a multirecord file containing three parameter records (INIT, RESTRT, and RECOVR) and seven job records (JOBNIP, JOBNS, JOBCS, JOBNVF, JOBTVF, JOBCOL, and JOBPUR). For the initial NAM startup, the operator should specify the parameter record INIT.

Subsequent entries of NAM from the console cause NAMI to use the parameter record RESTRT. The INIT record directs NAMI to route to the input queue jobs to start the programs NS, CS, NVF, TVF, and COLLECT. NIP will be started at the NAMI control point. If RBF has been installed, the INIT record also routes a job to start the program RBF.

INSTALLATION PARAMETERS

To assemble the following features into NAM, include directives of the form

```
*DEFINE name
```

on file USER for the NAM5 installation procedure.

NOTE

You should be thoroughly familiar with NAM operations before defining DEBUG and/or STAT.

name	Significance When Defined
DEBUG	Code to aid in debugging and maintenance in NIP and in PIP is generated. The following shows the effect of DEBUG on NAM components. • NIP, CS, NS, and NVF abort on certain error conditions. • CS, NS, and NVF are loaded with the debug version of AIP and produce network traces. • PIP hangs PPs for certain error conditions. • NIP uses internal trace buffers to trace messages sent and received and to trace subroutine and overlay calls.
IMS	Descriptive internal maintenance comments are included in the assembly and compilation listings.
NOPRU	Code for PRU interface is not generated. Specify if your site does not have batch processing (RBF).

name	Significance When Defined
STAT	Additional statistics-producing code is generated in NIP and AIP. With STAT defined, each time an application stops talking to the network, a terminal-to-application connection terminates, or an application-to-application connection terminates, statistical information is written to the NIP dayfile. After NIP terminates, the dayfile indicates the number of times each overlay was called and gives the statistics kept in common block STATTAB. The size of the job dayfile increases significantly when STAT is defined.
ZZDN	Code is generated to log all inbound or outbound messages between PIP and NIP in local file ZZZZZDN.

The following parameter is defined in deck INPARU. To make any changes to this parameter, place the appropriate Update directive on file USER when running the NAM5 installation job.

Parameter	Released Default Value	Significance
MAXNIP	20480	Maximum field length that NIP can reach. Range is 7680 to 131071.

Use the following formula to determine a value for MAXNIP for a particular configuration.

$$MAXNIP = 7000 + 340h + 6560a + m + k_1w_1 + \ldots + k_nw_n + 78b_1 + 142b_2 + 206b_3$$

Variable	Description
h	Maximum number of host nodes.
a	Maximum number of applications with up through eight application-to-application connections.
m	Maximum node number.
k_iw_i	Words per terminal. This value must be determined for each of the terminals configured in the local configuration file. It depends upon both the application block limit and the network block limit defined for each terminal and the type of terminal, as follows:

Variable	Description
k_i	Application block limit (ABL) and network block limit (NBL); one of the following values.

Value	Description
1	ABL $\leq$ 2 and NBL $\leq$ 2.
ABL - 1	ABL > 2 and NBL $\leq$ 2.
NBL - 1	ABL $\leq$ 2 and NBL > 2.
ABL + NBL - 2	ABL > 2 and NBL > 2.

Parameter	Released Default Value	Significance

Variable	Description

Variable	Description
w_i	Type of terminal; one of the following values.

Value	Description
20	Connected interactive terminal.
40	Connected batch terminal.
140	Active batch terminal.

Refer to the Network Definition Language Reference Manual for further information on ABL and NBL.

b_1	Number of 64-word PRU buffers (N1PRUB) allocated to NAM.
b_2	Number of 128-word PRU buffers (N2PRUB) allocated to NAM.
b_3	Number of 192-word PRU buffers (N3PRUB) allocated to NAM.

The N1PRUB, N2PRUB, and N3PRUB variables are NIP command parameters. Refer to NIP - Network Interface in this section.

Round MAXNIP up to the nearest multiple of 64.

NETWORK STARTUP MASTER FILE

The master file that NAM uses to start network processing consists of a set of text records that are either parameter records or job skeleton records. The master file is in standard NOS text format. You may create the file through XEDIT or a similar NOS utility. The master file provided on the NAM program library on REL12G can be modified through XEDIT or another text editor, or through Update directives against the NAM PL.

A comment can follow the parameters on all statements and directives.

TITLE Statement

The TITLE statement designates a record as a parameter record or a job skeleton record. It must be the first statement (following the name of the record) of every record in the master file.

The format is:

```
TITLE(type)title comment
```

Parameter	Significance
type	Type of record; use PARAM for parameter records and JOB for job skeleton records.
title	Text string of 1 through 50 characters that you can use to describe the purpose of the record.

Example:

    TITLE(PARAM) INIT - Initial network invocation

Parameter Records

Parameter records contain two kinds of directives that tell NAMI what parameters to substitute in job skeletons (PARAM directives) and what jobs to start (JOB directives). Each record can consist of a number of lines or directives up to 80 characters long, terminated by a zero byte.

Refer to figure 5-1 for an example of a parameter record.

```
INIT
TITLE(PARAM) INIT - INITIAL NETWORK INVOCATION.
.*
.*     THIS PARAMETER RECORD IS SELECTED WHEN THE NETWORK
.*     IS TO BE STARTED WITH FRESH LOADS OF THE FIRST
.*     LEVEL NPU(S), AND THEIR PREVIOUS CONTENTS
.*     ARE NOT TO BE DUMPED.
.*
.*
.*     1.  PURGE ALL PREVIOUS NETWORK DUMPS AND TRACES.
.*
.*     2.  LOCAL NPU(S) WILL BE STOPPED AND RELOADED
.*         WHEN THE NETWORK IS STARTED.
.*
.*     3.  LOCAL NPU(S) WILL NOT BE DUMPED WHEN THEY
.*         ARE INITIALLY LOADED.
.*
.*     4.  LOCAL NPU(S) WILL BE STOPPED IF THE HOST
.*         NETWORK FAILS.
.*
.*     5.  A FAILING NPU WILL TRIGGER HOST SUPERVISOR
.*         PROGRAM FIELD LENGTH DUMPS TO BE TAKEN AND
.*         PREVIOUS TRACE FILES TO BE PRESERVED.
.*
PARAM(NCFFN=NCFFILE) NETWORK CONFIGURATION FILE.
PARAM(LCFFN=LCFFILE) LOCAL   CONFIGURATION FILE.
PARAM(NLFFN=NLFFILE) NETWORK LOAD FILE (CCP).
PARAM(NIISTP=YES)    STOP NPU(S) AT HOST INITIALIZATION.
PARAM(NSFDP=NO)      INITIALLY LOADED NPU(S) NO DUMP.
PARAM(NIFSTP=YES)    STOP NPU(S) AT HOST FAILURE.
PARAM(NSRT=YES)      HOST DUMPS/TRACES ON NPU FAILURE.
PARAM(ZZMC=500)      MESSAGE COUNT BEFORE RELEASE TRACE FILE.
PARAM(ZZWHLCT=10)    MAX NUMBER OF AUTO RESTART OF PROGRAM.
.*
.*
JOB(JOBPUR,CO)    COLLECTOR JOB.
JOB(JOBNIP,NIP)   NAM (NIP) JOB.
JOB(JOBNVF,NV)    NVF JOB.
JOB(JOBCS,CS)     CS JOB.
JOB(JOBNS,NS)     NS JOB.
JOB(JOBTVF,TV)    TVF JOB.
.*
```

Figure 5-1. Example of Parameter Record

PARAM Directive

The PARAM directive is used in the parameter record to define keywords and values that are substituted for matching keywords in the job skeleton records. PARAM has the following format.

PARAM($keyword_1$=$value_1$,$keyword_2$=$value_2$,...$keyword_n$=$value_n$) comment

Multiple PARAM directives can appear in a parameter record.

The following rules apply to the PARAM directive.

- Embedded spaces are ignored.
- Keywords and values can contain only letters, digits, and asterisks.
- Keywords and values must be no longer than seven characters.
- If a keyword appears more than once, only the first definition applies.

Example:

When the following directive is present, every occurrence of the keyword NCFFN in any job skeleton record will be replaced by the string NCFFILE.

```
PARAM(NCFFN=NCFFILE)  Network configuration file.
```

JOB Directive

The JOB directive specifies the name of a job skeleton record and a code for the name of the network product that the job skeleton starts. A JOB directive must appear in every parameter record for each Network Host Products program that NAMI should start. The JOB directive has the following format.

JOB(name,type,ssname) comment

Parameter	Significance
name	Name of a job skeleton record.
type	First two or three characters of an application or program name, such as NS, CS, NIP, and so on.
ssname	Subsystem name if program is a subsystem (not required for NIP).

The rules for format of a PARAM directive apply to the JOB directive.

Example:

```
JOB(JOBNIP,NIP)  NAM (NIP) job.
```

Job Skeleton Records

Each job skeleton record contains the commands and input records required to start one network program. The record is in NOS job file format. In any of the commands or input record lines, any keyword (1 to 7 letters, digits, or asterisks, delimited by separators) may be a substitutable parameter. That is, if any keyword in the job skeleton record is identical to a keyword in the parameter record, NAMI substitutes the corresponding value from the parameter record for the keyword in the job skeleton. If a separator is an underscore, NAMI removes the underscore from the record when it makes the replacement.

In addition to substituting values for keywords defined in the parameter record, NAMI also substitutes variables known to it at the time it executes. These variables pertain to the startup master file currently in use and to the names of dump and trace files which NAMI creates with each new startup of NIP. Certain reserved keywords have been defined for use in the job skeleton records wherever one of the NAMI variables should be substituted.

Keyword	Meaning
CIN	Current network invocation number
OIN	Old network invocation number
UNM	Master file user name
PWM	Master file password
~~PNM~~ MFM	Master file name

The following keywords are defined by NAMI and should be used in the job skeleton record wherever the dump and trace files are to be referenced.

Keyword	Meaning
xxD0FIL xxD1FIL	Program dumps
xxD2FIL	Binary dumps of field length
xxL0FIL	Listable output
xxT0FIL	ZZZZZDN
xxS0FIL	ZZZZZSN

xx is the first two characters of the type from the JOB directive in the parameter record.

Job skeleton records must be constructed so that the files whose existence is assumed by the various programs are present.

NAMI ignores all comment cards in the job skeleton record of the form

.*text

NAMI writes an end-of-record in the submitted job when it encounters a card image of the following format in the job skeleton:

.EOR

Refer to figure 5-2 for an example of a job skeleton record.

```
JOBNIP
TITLE(JOB) JOBNIP - NIP RELEASE DEFAULT JOB SKELETON.
.*
.*        THE FOLLOWING PARAMETERS MUST BE SET IN THE PARAMETER RECORD,
.*
.*        NIISTP = STOP NPU(S) AT HOST INITIALIZATION.
.*
.*        NIFSTP = STOP NPU(S) AT HOST TERMINATION.
.*
USER(UNM,PWM)
NORERUN.
.*
.*   DEFINE FILES THAT RUN DATA WILL BE KEPT ON.
.*
.*        PFN         LFN           CONTENTS
.*
.*        NIDOFIL   OUTPUT        OUTPUT FROM JOB (DMP,DMD ETC).
.*        NID2FIL   ZZZZDMB       BINARY FIELD LENGTH DUMPS.
.*        NILOFIL   LIST          DAYFILE OF JOB.
.*
RETURN(OUTPUT)
DEFINE(OUTPUT=NIDOFIL,LIST=NILOFIL,ZZZZDMB=NID2FIL)
.*
.*   START NIP.
.*
SETJOB(UJN=NAM_CIN)
RFL(60000)
NIP(NIN=CIN,ISTP=NIISTP,FSTP=NIFSTP)
RFL(0)
.*
.*
 *   NIP NORMAL TERMINATION, PURGE ALL RUN DATA ON PERMANENT FILES.
.*
.*
PURGE(NIDOFIL,NID2FIL,NILOFIL/NA)
RETURN(OUTPUT)
WRITER(OUTPUT)
EXIT.
.*
.*        NIP FAILED.
.*
DISPLAY(EF)
NOTE(L,NR)+NIDA_CIN
DAYFILE(OP=F,L=L)
PACK(L)
COPYEI(L,LIST)
RETURN(OUTPUT)
WRITER(OUTPUT)
            -EOR-
```

Figure 5-2. Example of Job Skeleton Record

Network Host Product (NHP) Program Requirements

Job skeleton records for the CS, NIP, NS, and NVF jobs must each contain a command that calls the program that the job intends to run. These commands have the following format.

prog($keyword_1$=$value_1$,...$keyword_n$=$value_n$)

prog is the program name.
$keyword_i$=$value_i$ are order-independent parameters that are optional unless otherwise specified.

The files used by each program are listed following the description of each program command.

CS -- Communications Supervisor

The communications supervisor program (CS) requires the following command.

CS(MC=mc,NIN=nin)

Parameter	Description
MC=mc	Maximum message count; specifies the maximum number of messages to be accumulated in the debug log file before NETREL is called. No NETREL call will be issued if mc is 0. The default value is 500.
NIN=nin	Network invocation number; one- to three-digit decimal number assigned by NAMI when the network operation is started. This parameter is required.

CS requires the following files.

Name	Description
NCF	Network configuration file created by NDLP.
NRF1	Job record to be copied to the ZZZZZDN file by NETREL, that will determine the disposition of the network trace file, when NETREL is called on a periodic basis.
NRF2	Job record to be copied to the ZZZZZDN file by NETREL, that will determine the disposition of the network trace file, when NETREL is called as a result of an operator command or NPU failure.

NOTE

The default job skeleton for CS creates NRF1 and NRF2 from the input records of the CS job.

NIP -- Network Interface

The network interface program (NIP) requires the following command.

```
NIP(NIN=nin,ISTP=istp,FSTP=fstp,N1PRUB=n1prub,N2PRUB=n2prub,N3PRUB=n3prub)
```

Parameter	Description
NIN=nin	Network invocation number; one- to three-digit decimal number assigned by NAMI when the network operation is started. This parameter is required.
ISTP=istp	Option to stop all local NPUs during network host initialization. The default is NO. istp — Significance YES — Stop all local NPUs. NO — Do not stop local NPUs.
FSTP=fstp	Option to stop all local NPUs upon network host termination. The default is YES. fstp — Significance YES — Stop all local NPUs. NO — Do not stop local NPUs.
N1PRUB=n1prub	Number of 64-word buffers to be allocated during NIP initialization, from 0 through 12. The default value is 2.
N2PRUB=n2prub	Number of 128-word buffers to be allocated during NIP initialization, from 0 through 12. The default value is 6.
N3PRUB=n3prub	Number of 192-word buffers to be allocated during NIP initialization, from 0 through 12. The default value is 2.

Correct PRU buffer allocation increases the throughput of the host software for batch data and significantly reduces the host software's resource use when it is handling batch traffic. The buffers specified by the values n1prub, n2prub, and n3prub are shared between all drivers and all active PRU streams. The network configuration file allows you to select the number of PRUs to be transferred between the driver and the NPU.

Since performance is related to available buffers, correct PRU buffer allocation is important. In general, a PRU buffer can support from four through six active data streams at 9600 baud. The suggested device configuration is two 64-word PRU buffers (N1PRUB=2), six 128-word PRU buffers (N2PRUB=6), and two 192-word PRU buffers (N3PRUB=3) for each driver that supports the PRU data transfers. At least one PRU buffer of each size, up to the maximum UBZ or DBZ specified in NDL (refer to the Network Definition Language Reference Manual), must be allocated for each driver supporting PRU data transfers.

To determine if the PRU buffer allocation is correct, assemble the statistics feature (STAT) into NAM. The statistics feature causes the NAM dayfile to display at network termination the percentage of PRU buffer use. Ideally, the batch buffer use should be between 50 percent and 70 percent. To determine if the percentage of PRU buffer use is correct at your installation, load the network and run a typical remote batch load. At the completion of the run, terminate the network. Once the PRU buffer allocation is correct, you may want to recompile NAM with STAT turned off, because the statistics feature increases the size of the job dayfile.

NOTE

Leave PIP and its associated overlays on disk. Moving these overlays to central memory will not increase performance, since PIP copies its transient overlays to NAM's field length during its initialization.

NS -- Network Supervisor

The network supervisor program (NS) requires the following command.

```
NS(NIN=nin,FDP=fdp,RT=rt,MC=mc)
```

Parameter	Description
NIN=nin	Network invocation number; one- to three-digit decimal number assigned by NAMI when the network operation is started. This parameter is required.
FDP=fdp	Forced dump option.
RT=rt	Release trace file option.
MC=mc	Maximum message count; specifies the maximum number of messages to be accumulated in the debug log file before NETREL is called. No NETREL call will be issued if mc is 0. The default value is 500.

FDP=fdp:

fdp	Significance
YES	Within the first ten minutes of program execution, in the absence of other NPU dumping conditions, NPUs will be dumped before loading takes place.
NO	NPUs will not be dumped. The default is NO.

RT=rt:

rt	Significance
YES	NAM requests network supervisory programs CS and NVF to release their trace files whenever NS dumps an NPU. The default is YES.
NO	Trace files are not requested to be released.

NS requires the following files.

Name	Description
NLF	Network load file created by LFG (refer to the NOS 2 System Maintenance Reference Manual.
NCF, NRF1, and NRF2	Described under CS.

NVF — Network Validation Facility

The network validation facility requires the following command.

NVF(AL=arl,LL=lrl,MC=mc,NIN=nin)

Parameter	Description
AL=arl	Application retry limit. This parameter specifies the maximum number of invalid application connection request attempts an application is allowed before NVF considers the application to be breaching security and aborts the application. The value can range from 1 through 4. The default is 1.
LL=lrl	Login retry limit. This parameter specifies the maximum number of invalid login attempts a user is allowed before NVF considers the user to be breaching security and terminates the connection. The value can range from 1 through 4. The default is 4.
MC=mc	Maximum message count; specifies the maximum number of messages to be accumulated in the debug log file before NETREL is called. No NETREL call will be issued if mc is 0. The default value is 500.
NIN=nin	Network invocation number; one- to three-digit decimal number assigned by NAMI when the network operation is started. This parameter is required.

NVF requires the following files.

Name	Description
LCF	The local configuration file created by NDLP.
NRF1 and NRF2	Described under CS.

FILE PLACEMENT

Use the MOVEPF utility (refer to Step 13 - Move Files in section 2) to place the files NAM and NAMNOGO on the system user index (377777_8) and to place the local and network configuration files created with NDLP, the NAMI startup master file, and the load file created by LFG under the user name NETOPSX (user index 377772_8).

If RBF is to be installed, the RBF build should be completed before the NAMI startup master file is moved. The RBF build job modifies the startup master file so that RBF will be brought up automatically whenever NAM is brought up.

NOS — NETWORK OPERATING SYSTEM

The IAF interface is, by default, included when NOS is built. To build NOS without the IAF interface include the keyword NOIAF in the call to the procedure NOS.

OSTEXT - OPERATING SYSTEM TEXTS

No special information is needed to install OSTEXT.

PASCAL - PASCAL 170

Hardware requirements and special notes for installation of PASCAL 170 follow.

HARDWARE REQUIREMENTS

PASCAL 170 requires 70000_8 words of central memory for installation.

SPECIAL NOTES

Installation of PASCAL 170 from source code requires PASCAL (REL8C).

When you install PASCAL for the first time, you must perform a binary installation of PASCAL before you run the install job. For subsequent installations of the product, simply run the install job; no unusual action is necessary.

PASCAL provides the compiler and execution-time system.

PLI — PL/I VERSION 1

Information on PL/I hardware requirements and installation parameters follows.

HARDWARE REQUIREMENTS

PL/I requires 154000_8 words of central memory for installation.

Installation of PL/I from source code requires PL/I (REL14A), CCG (REL14B), and COMPASS (REL3A). PL/I provides the compiler and execution time system, CCG is the code generator, and COMPASS supplies COMPCOM and the common common decks.

Proper execution of programs compiled under PL/I requires FORTRAN Common Library 4 (both math and I/O), BAMLIB, and AAMLIB.

INSTALLATION PARAMETERS

The system text IPTEXT should contain parameter values consistent with the computer model and with the operating system on which the compiler is installed and executed. The MODEL micro in deck IPARAMS on REL1E must be set to the correct value for the computer model being used. Refer to TEXT and TEXTIO in this section.

PLITEXT and RTSTEXT on REL14A and CMPLTXT and CCGTEXT on REL14B select symbol definitions from IPTEXT for use by the PL/I compiler, code generator, and execution time system.

PMD4 — FORTRAN 4 POSTMORTEM DUMP UTILITY

Refer to FCL1, FCL2, and PMD4 - FORTRAN Common Library Version 4 with Postmortem Dump Utility in this section.

PMD5 — FORTRAN 5 POSTMORTEM DUMP UTILITY

Refer to FCL5 and PMD5 - FORTRAN Common Library Version 5 with Postmortem Dump Utility in this section.

QU3 — QUERY UPDATE VERSION 3

The installation tool SYNGEN, and the various common decks that reside on the DDL 3 program library, must be available for the installation of Query Update 3.

The QU3 installation procedure provides you with the option of installing the Information Management Facility Version 1 (IMF 1) interface. To install Query Update 3 with the IMF 1 interface, specify the IMF keyword in the call that executes the QU3 procedure. IMF 1 must be installed prior to installing Query Update 3 with the IMF 1 interface.

The installation of IMF 1 is necessary only if you select the IMF interface option. If you are not using IMF 1, omit the IMF keyword when installing Query Update 3.

To assemble code to interface with Sort/Merge 5 rather than Sort/Merge 4 specify SRT5 on the call to QU3. Sort/Merge 5 must be installed prior to installing Query Update 3 with the SRT5 interface.

Both the IMF and the SRT5 keywords can be specified on the call to QU3.

RBF - REMOTE BATCH FACILITY VERSION 1

To disable log file creation for Remote Batch Facility (RBF) 1, specify the keyword NOTRACE on the call that executes the RBF5 procedure.

The Remote Batch Facility requires the following command on the network startup file.

```
RBF2PO(MC=mc)
```

Parameter	Description
MC=mc	Maximum message count; specifies the maximum number of messages to be accumulated in the debug log file before NETREL is called. No NETREL call will be issued if mc is 0. This parameter is optional; no NETREL call will be issued if MC is omitted. If NETREL is to be called, a file NRF1 must be assigned to the control point before the command is executed. File NRF1 must contain a valid job record for writing to the ZZZZZDN file. The released RBF job skeleton record creates NRF1 from the job input record.

INSTALLATION PARAMETERS

To assemble various features into RBF, include directives of the form

```
*DEFINE name
```

on file USER for the RBF5 installation job.

name	Significance When Defined
DEBUG	Code to aid in debugging and maintenance is generated.
IMS	Descriptive internal maintenance comments are included in the assembly and compilation listings.
TRACE	Symbolic table dumps of RBF are written to file SPITOUT when RBF fails.

NOTE

You should be thoroughly familiar with RBF operations before defining DEBUG and/or TRACE.

The following parameters are defined in the common deck IP$COM. To make changes to these parameters, place appropriate Update directives on file USER for the RBF5 installation job.

Parameter	Released Default Value (Decimal)	Significance
SEARCHTIME	15	Time interval in seconds between scans of the output queue for remote batch files.†
RESUMETIME	20	Time interval in seconds between receipt of the last interactive message and the automatic switching of the terminal to batch mode; should be larger than SEARCHTIME.
REFRESHTIME	30	Refresh period in seconds for the RBF console queue displays when RBF is specified on the DISPLAY command; should be larger than RESUMETIME.
STATIONS	16	Maximum number of consoles.
TOTDEV	32	Maximum number of batch devices.
MAXFL	50000	If TRACE is not defined, maximum field length to which RBF expands when obtaining buffers.
MAXFL	100000	If TRACE is defined, maximum field length to which RBF expands when obtaining buffers.

FILE PLACEMENT

Execution of the RBF5 installation procedure requires a prior build of NAM and the existence of the NAM startup master file under the build user index. The RBF5 installation procedure modifies the startup master file. After RBF5 has completed, you can use the MOVEPF utility to move the startup master file to the system user index (377777_8).

VERIFICATION PROCEDURES

Use the following procedure to verify correct installation of RBF.

1. Prepare the following card deck. If necessary, supply the CHARGE command.

```
job command.
USER,username,password,familyname.
CHARGE,*.
NOTE.+1TOP OF PAGE+0ABCDEFGHIJKLMNOPQRSTUVWXYZ
6/7/8/9††
```

†These times are increased by approximately 10 seconds when the load on RBF is light and when most of RBF's field length is rolled out to disk.
††A multipunch in column 1 (or for HASP terminals, /*EOI in columns 1 through 5).

2. Initiate NAM as described under NAM5 - Network Access Method. (It is assumed that RBF is defined as an application in COMTNAP.)

3. Log in from the console of any network-supported terminal with at least one card reader and one line printer, specifying RBF as the application and any user name permitted to use RBF (refer to the Remote Batch Facility Reference Manual).† RBF responds with a header line giving the version of RBF and the date and time of login, followed by READY on the following line.

4. Enter DIS. RBF displays the status of the batch devices associated with the terminal.

5. Place the card deck prepared in step 1 into the input hopper of the card reader and initiate reading (described in the Remote Batch Facility Reference Manual).

6. Enter the following command at the terminal console.

   GO,CRn

Parameter	Description
n	Device number of card reader (from 1 through 7).

7. RBF reads the job in the card reader and responds with READY.

8. Enter DIS,PR at the terminal console. Because the system processes the job prepared in step 1 in little time, RBF should indicate that the job is in the print queue. If not, repeat this step until such an indication is received.

9. Ready the line printer, and enter the following command at the terminal console.

   GO,LPn

Parameter	Description
n	Device number of line printer (from 1 through 7).

10. RBF prints the job's output on the line printer. The output consists of a banner page, a page with TOP OF PAGE printed on the first line of the page and the alphabet on the third line, and a page containing the job dayfile.

11. Enter BYE to exit RBF and log off of the system. The system issues a message indicating the time that has elapsed since connection to RBF.

   RBF ENDED yy/mm/dd. hh.mm.ss

   RBF CONNECT TIME hh.mm.ss

---

†Follow all commands issued in this procedure by the message transmission key for the terminal class you are using (ETX or SEND for many batch terminal consoles).

RHF - REMOTE HOST FACILITY VERSION 1

The Remote Host Facility (RHF) links computer systems through a Loosely Coupled Network (LCN).

LCNLIB is required when you have site-developed RHF applications. LCNLIB is saved as a direct access file when you specify the keyword SAVLCN on the call to the installation procedure RHF.

RHF operations are enabled by the following two DSD entries.

```
ENABLE,RHF,n.
RHFffff.
```

n is the control point number and ffff is from zero through four alphanumeric characters. RHFffff can be a procedure file stored in an indirect access permanent file under the system user index (377777_8). If RHF operations are to be initiated automatically upon entry of the DSD AUTO command, have an ENABLE,RHF,n. entry (n is the control point number) in the IPRDECK. The entry point of the RHF subsystem is named RHF. Therefore, to make the DSD entry RHF., you need not store a procedure file named RHF under the system user index. However, you may use a procedure file named RHF. If you do, that file must contain a RETURN,RHF command to return the local file RHF before the call to RHF. For example,

```
.PROC,RHF,...
RETURN,RHF.
local site commands
RHF.
```

HARDWARE CONFIGURATION

RHF and its applications require the same minimum hardware configuration as NOS plus a minimum of one 380-170 Network Access Device (NAD).

INSTALLATION PARAMETERS

All necessary information for an installation to change the following parameters is included in the description for each parameter.

Parameter	Description
TIMEOUT	The time in seconds in which a response must be received by an application from the remote NAD/application before the connection is broken. Values may range from 1 through 1800 seconds. Default is 600. The current definition format (acceptable to both COMPASS and SYMPL) is:

```
1         11     18    24         34
#TIMEOUT  #DEF#  600D  #TIMEOUT   #600#;
```

The change deletion location is COMCAPR.131. The following is an example of a parameter change.

```
*D,COMCAPR.131
#TIMEOUT  #DEF#  400D  #TIMEOUT   #400#;
```

Parameter	Description

MAXRTRY The number of retries that an application will attempt to successfully complete a file transfer. Values may range from 1 through 50. Default is 10. The current definition format is:

```
1         11     18   24        34
#MAXRTRY  #DEF#  10D  #MAXRTRY  #10#;
```

The change deletion location is COMCAPR.74. The following is an example of a parameter change.

```
*D,COMCAPR.74
#MAXRTRY  #DEF#  20   #MAXRTRY  #20#;
```

MAXFILEXFR The maximum number of file transfers that the Facility Interface Program (FIP) allows for any one application. Values may range from 1 through 10. Default is 4. The current definition format is:

```
1      7               22
       DEF MAXFILEXFR  #4#;
```

The change deletion location is COMADEF.26 The following is an example of a parameter change.

```
*D,COMADEF.26
       DEF MAXFILEXFR  #5#;
```

FETBUFSIZE The number of words assigned to buffer space for each file transfer. Values may be zero or greater, but FIP will override low values.

FETBUFSIZE	Assigned (binary)	Assigned (coded)
0 to 532	532	992
532 to 992	532 to 992	992
992 and up	992 and up	992 and up

The default value for FETBUFSIZE is 3187, which corresponds to about 49 PRUs. Values larger than 6400 (98 PRUs) do not increase transfer rates appreciably, and make job swapping more likely because of the increased central memory required.

The change deletion location is C5471FCB.96. The following is an example of a parameter change:

```
*D,C5471FCB.96
       DEF FETBUFSIZE #3200#;
*C COMAFET,PRSTFIP,XFRINIT
```

QTF Initiation Procedure Parameters

RHF uses the system procedure QTF to initiate the RHF application QTF (queue file transfer facility). The parameter used by the procedure controls the interval length between QTF executions.

The default procedure definition is:

```
.PROC,QTF,DEL=1.
```

Parameter	Description
DEL=del	Time in minutes between job executions (0 < del $\leq$ 2048).

The change deletion location is QTFPROC.3. The following are examples of a parameter change.

```
*D,QTFPROC.3
.PROC,QTF,DEL=3.
```

MLTF Initiation Procedure Parameters

RHF uses the system procedure MLTF (maintenance error logging transfer facility) to initiate periodic NAD error logging. (Routine NETLOG in the MLTF procedure does the actual recording of NAD errors in the binary maintenance log (BML).) The parameter used by the MLTF procedure controls the interval length between MLTF executions.

The default procedure definition is:

```
.PROC,MLTF,DEL=60.
```

Parameter	Description
DEL=del	Time in minutes between MLTF executions (0 < del < 2048).

The change deletion location is MLTFPROC.3. The following is an example of a parameter change:

```
*D,MLTFPROC.3
.PROC,MLTF,DEL=15.
```

RHF - CONFIGURATION FILE GENERATION

You must define the RHF configuration including all NADs, applications, and logical identifiers (LIDs) or physical identifiers (PIDs) to be used by or accessible to RHF. You may define multiple configurations. Each RHF in the network must have its own definition of the configuration, which will be different from the definition of the configuration for other RHFs in the network. Use the RCFGEN utility to define these configurations.

The RCFGEN utility reads configuration definition statements to create a direct access permanent file under user name SYSTEMX. RHF then uses the permanent file as the network description and can properly access the network.

When RHF is started, it expects to find a local configuration file named RCFILE. If it does not find this file, RHF will attach the configuration file using permanent file name RCFMxx, where xx is the machine identifier.

The following command calls the RCFGEN utility.

```
RCFGEN,I=lfn,L=lfn,O=lfn.
```

The parameters are order-independent and optional. If omitted, the defaults are used.

Parameter	Description
I=lfn	Specifies the local file from which the RHF configuration statements are to be read. The default is file INPUT.
L=lfn	Specifies the local file to which all listable output is to be written. The default is file OUTPUT.
O=lfn	Specifies the local file to which the configuration tables are to be written. The default is file RCFILE.

The input to RCFGEN consists of network configuration statements. The syntax of input statements conforms to COMPASS macro requirements. Configuration statements (except local and remote NAD configuration statements) must not start before character position 3.

Use the following configuration definition statements.

Statement Name	Description
LNDR	Defines the maximum number of local NAD drivers (NDRs) allowed to execute at one time.
NPID	Defines the physical ID of a remote mainframe.
NLID	Defines the logical ID of a remote mainframe.
PATH	Defines the paths to a remote mainframe through the LCN Network.
RNAD	Defines the addressing information necessary to access a remote NAD.
LNAD	Defines information necessary to address local NADs.
APPL	Defines application programs that are allowed to access RHF.

Statement Name	Description
DEBUG	Defines debug parameters.
CHARGE	Defines the charge that is transferred to a user control point for each RHF call.

The NLID and PATH statements must be associated with a given physical mainframe (NPID statement). The following structure is required of these statements when defining a network:

1. NPID statement.

2. All lids (NLID statement) associated with the preceding NPID.

3. All paths (PATH statement) associated with the preceding NPID.

Configuration definition statements must be specified in the following order to properly define a network.

1. LNDR statement (if necessary).

2. Application programs.

3. Repeat NPID, NLID, and PATH statements until all portions of the network are defined.

4. LNAD statements to define local NADs.

5. RNAD statements to define hardware addressing of the remote NADs.

6. DEBUG and CHARGE statements (order-independent).

NOTE

At least one of each configuration statement is required in the configuration file with the exceptions of the LNDR, DEBUG, and CHARGE statements. Defaults are specified in the individual statement descriptions.

NETWORK CONFIGURATION STATEMENTS

Following are the network configuration statements and their descriptions.

Remote Mainframe Definition

Each remote mainframe definition requires 3 words in RHF's field length. To define a remote mainframe, enter:

```
NPID   PID=pid,ENABLED=status,MFTYPE=type
```

Parameter	Description
PID=pid	Physical identifier of the remote mainframe (required); must be unique combination of three alphanumeric characters.
ENABLED=status	Availability of the mainframe identified by PID. Values may be YES or NO. Default is YES.
MFTYPE=type	Mainframe type (required); 1- to 7-alphanumeric character string. For example, you could use NOS1, NOS2, NOS/BE, or CY200. This parameter is used only to display information to the operator on the ID display.

LID Definitions for the Remote Mainframe

Every two LIDs defined for a remote mainframe require one word in RHF's field length. To define a LID for a remote mainframe, enter:

```
NLID   LID=lid,ENABLED=status
```

Parameter	Description
LID=lid	Logical identifier for the mainframe identified by the last PID definition; three alphanumeric characters. The lid may be the same as the PID for the last NPID configuration statement. The LID parameter is required.
ENABLED=status	Availability of the mainframe using this LID and identified by PID; values may be YES or NO. Default is YES.

NOTE

At least one NLID statement is required for each PID defined.

Path Definitions to the Remote Mainframe

Every path defined for a remote mainframe requires two words in RHF's field length. To define a path for a remote mainframe, enter:

```
PATH    ENABLED=status,LT=tttt,RT=rrrr,RNAD=raddr,LNAD=laddr,AC=aaaa
```

Parameter	Description
ENABLED=status	Availability of the path when RHF is initialized. Values may be YES or NO. Default is YES.
LT=tttt	Local trunk control units (TCUs) enabled; a four digit non-zero binary number indicating the network trunk connections for the local NAD (required).
RT=rrrr	Remote TCUs enabled; a four digit non-zero binary number indicating the network trunk connections for the remote NAD (required).
RNAD=raddr	Symbolic address of the remote NAD entry for this path, referenced in the RNAD statement (required).
LNAD=laddr	Symbolic address of the local NAD entry for this path referenced in the LNAD statement (required).
AC=aaaa	Four-digit hexadecimal access code for the remote NAD. Default is zero.

NOTE

At least one PATH statement is required for each PID defined.

Local NAD Definitions

Each local NAD definition requires two words in RHF's field length. To define a local NAD, enter:

```
laddr    LNAD    CH=ch,MAXNDRS=n,DEDICATE=status
```

Parameter	Description
laddr	Symbolic address referenced in a preceding PATH statement. This parameter is required and must begin in character position one or two.
CH=ch	Channel (octal) that NAD is on.

Parameter	Description
MAXNDRS=n	Maximum number (1 through 3) of NAD drivers that may be assigned at one time to this NAD. Default is 1.
DEDICATE=status	Dedicated channel indicator; specifies whether the driver will always hold the NAD channel reservation between consecutive blocks of one I/O request. Values may be YES or NO. Default is YES. YES should always be specified unless some non-CDC driver requires high performance access to the NAD channel.

Remote NAD Definitions

Every remote NAD defined requires one word in RHF's field length. To define a remote NAD, enter:

```
raddr      RNAD    ND=nn,DD=d,LOG=status
```

Parameter	Description
raddr	Symbolic address referenced in a preceding path statement. This parameter is required and must begin in character position 1 or 2.
ND=nn	Two-digit hexadecimal remote device address indicating the address of the remote NAD. Default is zero.
DD=d	One-digit hexadecimal destination device address indicating the exit port of the remote NAD. Default is zero.
LOG=status	Error logging indicator; specifies if remote NAD trunk errors are to be recorded by MLTF in the mainframe's error log. Values may be YES or NO. Default is NO.

Application Definitions

Each application definition uses additional RHF field length. This additional field length is equal to mxcopys*(5+3*mxcons) CM words. To define an RHF application, enter:

```
APPL    NAME=name,ENABLED=status,MXCONS=mxcons,MXCOPYS=mxcopys,SVR=s,ASTART=a
```

Parameter	Description
NAME=name	Application name; 1- through 7-alphanumeric characters where the first character must be alphabetic (required).
ENABLED=status	Availability of the application when RHF is initiated. Values may be YES or NO. Default is YES.
MXCONS=mxcons	Maximum simultaneous connections the application may have. The maximum value is 127. Default is one.

Parameter	Description
MXCOPYS=mxcopys	Maximum number of simultaneously active copies of this application that are allowed. Maximum value is 127. Default is one.
SVR=s	Servicer program indicator; specifies if the application is a servicer, that is, an application program that is started automatically by RHF upon request of an application program on another host. Values may be YES or NO. Default is NO.
ASTART=a	Application startup indicator; specifies whether the application is started when RHF is initiated or when the operator enables the application. Values may be YES or NO. If NO is specified, the application is started by the user through QTF or MFLINK, or for a servicer application, by a request from a remote application. Default is NO.

In determining the number of allowed connections and copies of an application, note that each NAD has a maximum of 127 active connections. This number is restricted to 35 during NAD controlware loading but may be increased by modifying the appropriate NAD controlware load parameters in BCLOAD.

NOTE

When defining the APPL statements you must follow certain restrictions for system-supplied applications QTF, QTFS, PTF, PTFS, and MLTF. The maximum simultaneous connections should be set to one (default) for QTFS, PTF, PTFS, and MLTF. The maximum number of simultaneously active copies should be set to one (default) for QTF and MLTF. The maximum simultaneous connections should not be set greater than four unless installation parameter MAXFILEXFR is increased for QTF/FIP. The SVR=YES parameter must be specified for QTFS and PTFS and must not be specified for QTF, PTF, or MLTF (either SVR=NO or default). ASTART=YES should be specified for both QTF and MLTF.

Definition of Maximum NDRs Allowed for all NADs

The maximum number of NAD drivers is the maximum number of PPs that may contain a NAD driver at one time, regardless of the number of NADs and the number of drivers allowed per NAD.

To define the maximum NDRs allowed for all NADs, enter:

```
LNDR   MAXNDRS=nn
```

Parameter	Description
MAXNDRS=nn	Maximum number of PPs that may contain NDR at any one time. Default is one. The maximum value is the number of PPs available for NAD drivers. nn must not be less than the largest value specified for MAXNDRS on any local NAD (LNAD) statement.

DEBUG Statement

The DEBUG statement controls the manner in which RHF uses queue entries.

```
DEBUG   TRACE=t
```

Parameter	Description
TRACE=t	Values may be YES or NO. NO specifies that RHF trace is off. When RHF trace is off, queue entries freed by RHF are placed at the top of the empty queue and immediately reused. When NO is specified, RHF uses a slightly smaller amount of CP time. Default is NO. YES specifies that RHF trace is on. When trace is on, queue entries are reused only after all queue entries ahead of it have been used. Also, when trace is on, analysis of an RHF dump and resolution of the associated RHF problem may be facilitated.

UCP Charge Definition for an RHF Call

The CHARGE statement specifies the amount of system resources a user control point (UCP) is charged for an RHF call. RHF distinguishes two different types of calls; those that require a large amount of processing time and those that require a small amount of processing time. RHF charges the UCP more for calls that require a large amount of processing time.

```
CHARGE   TYPE=n,CPA=cpa,CPB=cpb,IO=io,CMFL=cm,PP=pp
```

Parameter	Description
TYPE=n	Type of call for which the charge is being specified. Value may be 1 or 2. 1 requires small amount of RHF processing time. 2 requires large amount of RHF processing time.
CPA=cpa	Time to be charged for central processor cpa (decimal milliseconds). Default is 2 milliseconds for a type 1 call, 10 milliseconds for a type 2 call.

Parameter	Description
CPB=cpb	Time to be charged for central processor cpb (decimal milliseconds). Default is zero.
IO=io	I/O time to be charged (decimal milliseconds). Default is zero.
CMFL=cm	CM field length (octal)/100B to be charged. Default is 10.
PP=pp	PP time to be charged (decimal milliseconds). Default is zero.

A CHARGE statement is not required. You may, however, enter two CHARGE statements: one for type 1 calls, a second for type 2 calls.

Figure 5-3 illustrates a sample LCN network and is the basis of the configuration statements in examples 1 and 2.

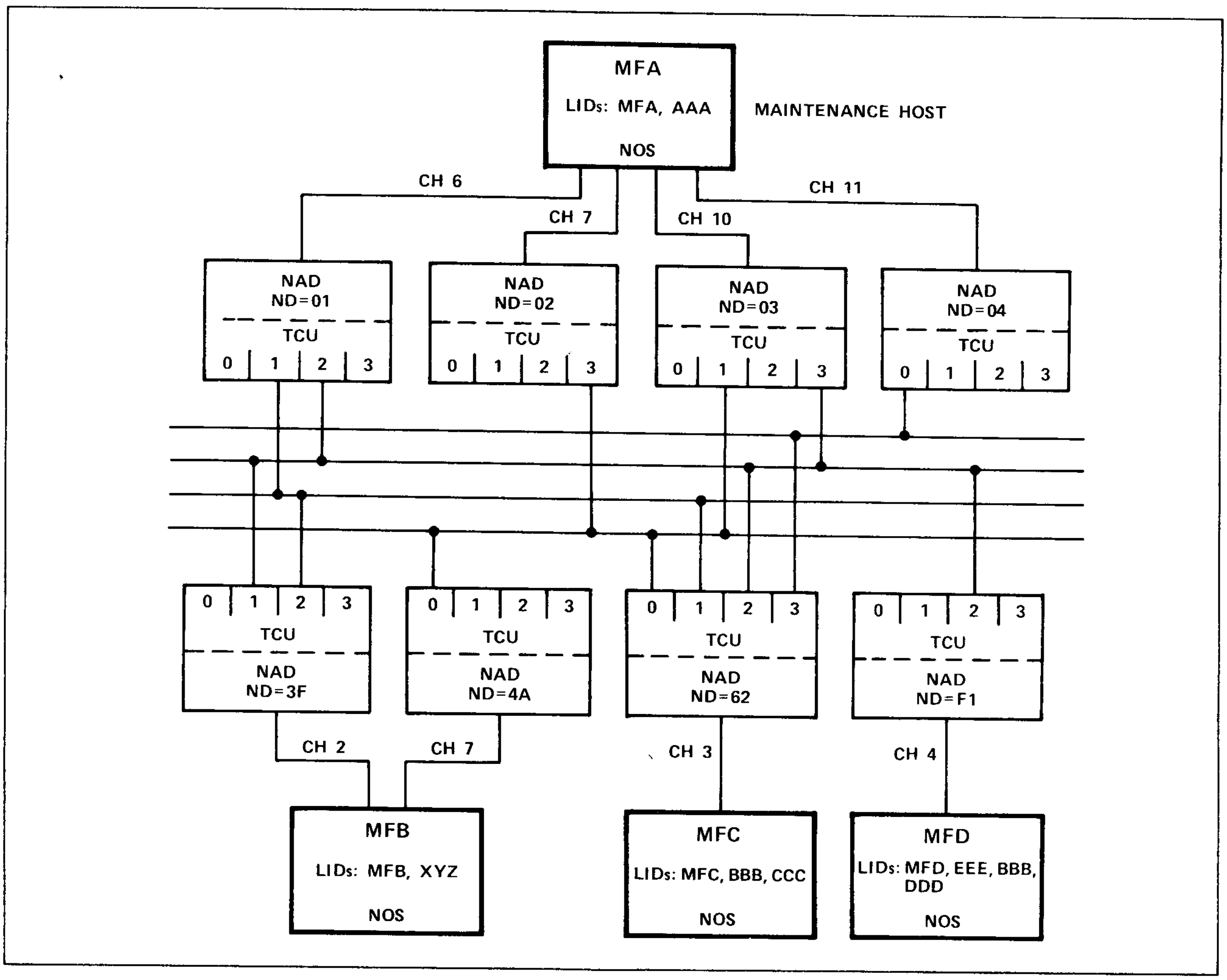

Figure 5-3. Sample LCN Network

Example 1. The following is a sample configuration for the RHF on mainframe MFA in figure 5-3.

```
        LNDR    MAXNDRS=3
        APPL    NAME=QTF,MXCONS=4,ASTART=YES
        APPL    NAME=QTFS,MXCONS=1,MXCOPYS=8,ASTART=NO,SVR=YES
        APPL    NAME=PTF,MXCONS=1,MXCOPYS=8,ASTART=NO
        APPL    NAME=PTFS,MXCOPYS=8,SVR=YES
        APPL    NAME=USRAP,MXCONS=6,ENABLED=NO
        APPL    NAME=MLTF,ASTART=YES

        NPID    PID=MFB,ENABLED=YES,MFTYPE=NOS2
        NLID    LID=MFB
        NLID    LID=XYZ
        PATH    ENABLED=YES,LT=0110,RT=0110,RNAD=RN2,LNAD=LN1,AC=FOFO
        PATH    ENABLED=YES,LT=0001,RT=1000,RNAD=RN1,LNAD=LN2,AC=FOFO
        PATH    ENABLED=NO,LT=0001,RT=0100,RNAD=RN2,LNAD=LN3,AC=FOFO
        PATH    ENABLED=YES,LT=0100,RT=1000,RNAD=RN1,LNAD=LN3,AC=FOFO

        NPID    PID=MFC,MFTYPE=NOS2
        NLID    LID=MFC
        NLID    LID=BBB,ENABLED=NO
        NLID    LID=CCC
        PATH    ENABLED=NO,LT=0110,RT=0110,RNAD=RN3,LNAD=LN1,AC=FOFO
        PATH    ENABLED=NO,LT=0001,RT=1000,RNAD=RN3,LNAD=LN2,AC=FOFO
        PATH    LT=0101,RT=1010,RNAD=RN3,LNAD=LN3,AC=FOFO
        PATH    LT=1000,RT=0001,RNAD=RN3,LNAD=LN4,AC=FOFO

        NPID    PID=MFD,MFTYPE=NOS2
        NLID    LID=MFD
        NLID    LID=EEE
        NLID    LID=DDD
        NLID    LID=BBB
        PATH    LT=0010,RT=0010,RNAD=RN4,LNAD=LN1,AC=FOFO
        PATH    LT=0001,RT=0010,RNAD=RN4,LNAD=LN3,AC=FOFO

    LN1 LNAD    CH=6
    LN2 LNAD    CH=7
    LN3 LNAD    CH=10
    LN4 LNAD    CH=11

    RN1 RNAD    DD=0,ND=4A,LOG=YES
    RN2 RNAD    DD=0,ND=3F,LOG=YES
    RN3 RNAD    DD=0,ND=62,LOG=YES
    RN4 RNAD    DD=0,ND=F1,LOG=YES
```

Example 2. The following is a sample configuration for the RHF on mainframe MFD in figure 5-3.

```
        LNDR     MAXNDRS=2

        APPL     NAME=QTF,ASTART=YES,MXCONS=4
        APPL     NAME=QTFS,MXCOPYS=4,SVR=YES
        APPL     NAME=PTF,MXCOPYS=6
        APPL     NAME=PTFS,MXCOPYS=6,SVR=YES
        APPL     NAME=MLTF,ASTART=YES

        NPID     PID=MFA,MFTYPE=NOS2
        NLID     LID=MFA
        NLID     LID=AAA
        PATH     RNAD=RN01,LNAD=LN1,LT=0010,RT=0010,AC=F0F0
        PATH     RNAD=RN03,LNAD=LN1,LT=0010,RT=0001,AC=F0F0

        NPID     PID=MFB,MFTYPE=NOS2
        NLID     LID=MFB
        NLID     LID=XYZ
        PATH     RNAD=RN3F,LNAD=LN1,LT=0010,RT=0100,AC=F0F0

        NPID     PID=MFC,MFTYPE=NOS2
        NLID     LID=MFB
        NLID     LID=BBB
        NLID     LID=CCC
        PATH     RNAD=RN62,LNAD=LN1,LT=0010,AC=F0F0

    LN1 LNAD     CH=4,MAXNDRS=2

    RN01     RNAD     ND=01,LOG=NO
    RN03     RNAD     ND=03,LOG=NO
    RN3F     RNAD     ND=3F,LOG=NO
    RN62     RNAD     ND=62,LOG=NO
```

LOGICAL IDENTIFIER DEFINITION AND USE

The Logical Identifier (LID) is the identifier used to refer to a mainframe. A user refers to an LID on an MFLINK command, a ROUTE command, or a Job command.

For a successful access to or from another mainframe, both the RHF configuration (including LIDS) and the LID table in CMR must be set up correctly. The other (remote) mainframe likewise must have its configuration and LID tables set up properly to receive or generate a successful network access.

For either QTF or MFLINK to initiate access to a remote mainframe, the LID used must be defined and enabled in the RHF configuration table. Likewise that LID must be defined, enabled, and must have the linked attribute set in the LID table in CMR. This can be done either through the IPRDECK LID entries or through the operator utility LIDOU (refer to the NOS 2 Operator/Analyst Handbook). When QTF initiates access to a remote mainframe, the LID in the CMR LID table should not have the host attribute set. If it does, the queue file will execute or be output locally and will not be transferred to the remote mainframe.

For either QTFS or PTFS to respond to a remote mainframe request for file transfer, the remote mainframe's physical identifier (PID) must be defined and enabled in RHF's configuration table as a valid PID for that mainframe. Likewise the LID specified on the remote mainframe must be defined and enabled, and must have the host attribute set in the CMR LID table.

SPECIAL LOOPBACK CAPABILITY

A special loopback capability is available on RHF for the use of both QTF and MFLINK. This capability is intended primarily for test purposes, but it may be used for other purposes as desired. The loopback capability allows a file to be sent from the local mainframe out to a NAD and back to the local mainframe instead of to a remote mainframe. To use this capability, the LIDs used to specify loopback must be defined properly both in RHF's configuration table and in the CMR LID table.

To allow loopback, a PID entry (and associated LID and PATH entries) must be defined to match the local mainframe (Mxx, where xx is the machine id) in RHF's configuration. LIDs to be used for loopback should follow. Next, the loopback path should be defined. The remote NAD specified on the PATH definition should have, as an address, the address of the local NAD.

The LID to be used for MFLINK loopback, in addition to being in RHF's configuration, also must be defined in the CMR LID table as enabled, and the host and linked attributes must be set.

The LID to be used for QTF loopback, in addition to being in RHF's configuration, also must be defined in the CMR LID table as enabled and linked. The host attribute must not be set or the loopback will not occur.

NOTE

The MFLINK and QTF loopback lids must be different.

Example: In the following RHF configuration file and IPRDECK entries, LBK is the loopback lid for MFLINK and LB2 is the loopback lid for QTF. The NAD on M42 is on channel 7 and has NAD address 1F.

```
      NPID    PID=M42,MFTYPE=NOSHOST
      NLID    LID=LBK
      NLID    LID=LB2
      PATH    LNAD=LN1,RNAD=RN1,LT=0001,RT=0001,AC=F0F0

LN1   LNAD   CH=7

RN1   RNAD   ND=1F
```

IPRDECK entries:

```
LID=LBK,H,L,V.
LID=LB2,L,V.
```

SORT — SORT/MERGE VERSION 4

Hardware requirements and installation parameters for Sort/Merge 4 follow.

HARDWARE REQUIREMENTS

If you use the tape sort option, polyphase requires three additional magnetic tape units and balanced requires four.

INSTALLATION PARAMETERS

Whether Sort/Merge 4 uses the CMU hardware depends upon the IP.CMU parameter (refer to TEXT and TEXTIO in this section) from IPTEXT. To override this parameter, make the following changes on file USER.

To install Sort/Merge 4 without CMU code (default):

```
*I,FEAT64.42

BDP.INST     EQU   BDP.NO
```

To install Sort/Merge 4 with CMU code:

```
*I,FEAT64.42

BDP.INST     EQU   BDP.YES
```

SORT5 — SORT/MERGE VERSION 5

No special information is needed to install Sort/Merge 5.

SYMPL — SYMPL VERSION 1

No special information is needed to install SYMPL.

TAF - TRANSACTION FACILITY VERSION 1

Establish a user name, user index, and password for TAF when creating or updating the validation files with MODVAL (refer to the NOS 2 System Maintenance Reference Manual). Set the installation parameters USNM, TRUI, and PWDM (described in the following paragraphs) to agree with the values established by MODVAL. For an overview of the TAF installation process, refer to the installation overview appendix in the TAF Reference Manual.

TAF uses NAM, so you must also install NAM. To install TAF, you must specify the keyword TAF on the call that executes the procedure COMBINE and later in the installation process you must execute the TAF installation procedure. You can install TAF with or without the trace feature. To get the trace feature, specify DEBUG on the call to the installation procedure TAF.

SEE ERRATA

After installing TAF on the deadstart tape, but before using it, create a task library permanent file containing the following required tasks, which are on release tape REL12C.

Task	Description
ITASK	Initial task.
BTASK	Task that recovers transactions initiated by BTRAN.
CRMTASK	Task that formats TAF/CRM Data Manager file status displays.
CTASK	Task to recover transactions using the TAF/CRM Data Manager.
RTASK	Task to recover terminals. RTASK may be on the task library permanent file or on database libraries.
RCTASK	Task that recovers CDCS transactions.
KDIS	TAF K display driver.
LOGT	Task to log out transaction terminal from TAF.
MSABT	Diagnostic generator for abnormally terminating tasks.
OFFTASK	Inactive task controller.
SYSMSG	Message task for system origin messages.
XTASK	Execute named task.

To create a task library containing these tasks, run a job similar to the following:

Job	Comments
job command.	
USER,username,password,familyname.	Use the user name and password previously assigned to TAF.
CHARGE,*.	
ATTACH,OPL/UN=LIBRARY.	User must have permission to access OPL. This job assumes that RELSOPL is saved as OPL under the user name LIBRARY.
MODIFY.	
COMPASS,I,S=NOSTEXT,L=0.	
DEFINE,TASKLIB/CT=PU,M=W.	
LDSET,LIB=TRANLIB,LIB=TRANF4.	
LOAD,LGO.	
NOGO,LGOB.	
LIBTASK,CR,Z.+/XTRAN,T1=XTASK.	Transaction XTRAN provides the capability of scheduling any task in the local or system task library. Enter EX. from a terminal to schedule XTRAN.

```
--eor--
*EDIT   ITASK,KDIS,LOGT,MSABT,OFFTASK,SYSMSG,XTASK
*EDIT   BTASK,CTASK,RTASK,RCTASK,CRMTASK
--eoi--
```

If TAF will be used in a multimainframe complex, the system does not allow concurrent access to the same data base. A copy of TAF in each computer must have its own user name/user index or default family.

FILE PLACEMENT

The call to installation procedure TAF compiles TAF and moves TAF binaries and the system procedure TAFPRC to the file PRODUCT and subsequently to the deadstart tape. To initiate TAF operations, the operator enters the DSD commands:

```
ENABLE,TAF,n.
TAFffff.
```

n is the control point at which TAF is to run. The name of the TAF procedure file and the DSD command to initiate TAF processing must be the same and must be of the form TAFffff (ffff is from zero through four alphanumeric characters). If the system is to initiate TAF operations automatically when the operator enters the DSD command AUTO, the procedure file must have the three-character name TAF, and TAF must be enabled in the IPRDECK. TAFffff must be either an indirect access permanent file stored under the system user index (377777_8) or a system resident procedure.

For information on activating the interface between COBOL 5 and TAF, refer to COBOL5 in this section.

INSTALLATION PARAMETERS

Unless otherwise specified, the following parameters are defined in deck COMKIPR. These parameters specify the charge and project numbers, password, and user index for TAF. They also specify the user name under which TAF runs.

Parameter	Released Default Value	Significance
CGNM	A null micro	Micro whose string specifies the charge number for TAF; used when a dump is performed. If CGNM is null, no CHARGE command is issued, and the user name specified by USNM must not require a CHARGE command.
PJNM	A null micro	Micro whose string specifies the project number for TAF.
PWDM	TAFPASS	TAF password.
TRUI	16B	User index for TAF.
USNM	KB100DC	Micro whose string specifies the user name under which TAF runs.

The following parameters specify the default initialization K display options.

Parameter	Released Default Value	Significance
ECSFL	0	Extended memory field length/1000_8. ECSFL cannot be less than 0 nor greater than 400_8.
NCMB	40	Actual number of communication blocks allowed in the subsystem. Communication blocks hold incoming terminal input. This parameter can be changed by the initialization command K.CMB, but it cannot be less than 19 nor greater than 40.
NSCP	31	Maximum number of subcontrol points. It cannot be less than 2 nor greater than 31.
SCMFL	376600B	Maximum field length. SCMFL cannot be less than 40000_8 nor greater than 376600_8.
TLFM	TASKLIB	Micro whose string specifies the system task library file name.

The following parameters, defined in deck TAF, specify the default DSDUMP parameters. The user can override the parameters specified on CMDUMP requests with a task.

Parameter	Released Default Value	Significance
DEXP	1	Exchange package dump flag: Value — Description 0 — Exchange package is not dumped. 1 — Exchange package is dumped.
DFWA	0	First word address in octal for task dump.
DLWA	100000B	Last word address in octal for task dump.
DORC	BCOT	Origin code.
DORT	0	Output disposition (corresponds to OQ parameter on DSDUMP/CMDUMP requests): Value — Description 0 — Local batch output queue. 1 — Remote batch output queue. 2 — Direct access permanent file. Refer to the TAF Reference Manual for further information.

Parameter	Released Default Value	Significance
DSQID	0	Batch identification (ID) code for output of jobs entered in the input queue by the task SUBMT request. The system assigns this ID to the output from jobs containing a SETJOB,DC=DF command. DSQID ranges from 0 through 67_8.

The following parameters specify default time dependencies. Although these values are expressed in milliseconds, they are accurate to only 1 second.

Parameter	Released Default Value	Significance
CORTL	1*1000	How often TAF checks to see if memory can be released to the system.
ITRTL	1500	Time to wait for input before rollout of transaction executive field length. ITRTL is defined in deck TAF.
RRTTL	1*1000	Time allowed to elapse before evicting a reusable task.
TACTL	2*60*1000	Time allowed to elapse between TAF receiving any input and TAF generating a call to ITASK. TACTL is defined in deck TAF.
TROTL	10*60*1000	Duration of rollout. TROTL is defined in deck TAF.
DMMTL	4	Time allowed to elapse between calls to the data manager(s).
TSKTL	120	Task time slice in milliseconds.

The following parameters, defined in deck TAF, specify default task rollout parameters.

Parameter	Released Default Value	Significance
DWITL	8*60	Time in seconds that a task is allowed to wait for terminal input before aborting. The user can override this parameter with the WAITINP request.
NESTL	16	Nest limit for CALLRTN (must be less than 64).
RTDNL	2*1000	Number of milliseconds a task is allowed to remain in memory waiting for a CALLRTN to complete.

The following parameters specify other default TAF installation parameters.

Parameter	Released Default Value	Significance
DTSTL	16	Number of time slices for a task. The user can override DTSTL for an individual task with the ITL request. DTSTL is defined in deck TAF.
IPTAR	1	Automatic recovery flag:

Value	Description
0	Automatic recovery is disabled.
1	Automatic recovery is enabled.

If recovery is disabled, the following requests will not be honored in recovery mode.

Request	Comments
CALLTRN	Transactions can be scheduled, but input is not logged to the communication recovery file (CRF).
RERUN	
RGET	
RPUT	
RSECURE	
SECURE	
TINVOKE	
TSTAT	Except for the keywords USER and NEXT.
WSTAT	Except for the keywords STEP (=8 or =9) and USER.

Parameter	Released Default Value	Significance
IPTST	500	Number of terminals that can access TAF. IPTST must be greater than 0 and less than 4095.
NTSB	50	Number of entries in the NTSB table that is used to hold entries for which work is queued; defined in deck TAF.

Parameter	Released Default Value	Significance
RECDF	0	Default user recovery flag: **Value** / **Description** 0 — User recovery is enabled. 1 — User recovery is disabled.
DTYM	DI	Micro whose string specifies the device type for journal files.
IFL=	150000B	Initialization field length; defined in deck TAF. This value must be large enough to load the Application Interface Program required for NAM interface, and the desired data managers and various tables required by TAF during initialization. If the message MEMORY OVERFLOW DURING INITIALIZATION is issued, either increase IFL= or decrease the data bases, the number of data manager buffers, or the number of communication blocks.
INRBL	1+63	Maximum length of the intercontrol point-request-receiving buffer; includes 1 header word plus 63 words for input. INRBL is defined in deck TAF.
MAXJL	2500	Maximum word count on one journal request to any journal file, including header words; defined in deck TAF.
MAXRA	500B	Task limit for RA+1 requests; defined in deck TAF.
MAXTO	6*MAXWS	Maximum number of words task can send to the communication subsystem. Equaling or exceeding this value causes the task to abort.
MAXWS	409+1	Number of words SEND can transmit plus 1. Exceeding this value causes the task to abort.
TLDL	TLDLE*10	Amount of space to reserve for added tasks in the TAF-resident copy of the directory of each task library attached by TAF. This space can be used when TAF is informed of a task library change through the LIBTASK TT option. The value of the symbol should be a multiple of the size of a task library directory entry (TLDLE, currently 3). The default value allows space for 10 (TLDL/TLDLE) additional tasks. If more than TLDL/TLDLE tasks are added by the TT option, only the first TLDL/TLDLE tasks can be executed. The next time TAF is reinitialized, however, all the tasks added via the TT option will be available to be executed. TLDL is defined in deck COMKTLD.

The following parameters are used with the TAF/CRM Data Manager.

Parameter	Released Default Value	Significance
AOBFL	31	Output queue length.
AIBFL	31	Input queue length.
CMAXDB	31	Number of CRM data bases.
CMDM	31	Number of concurrent TAF/CRM Data Manager requests and the number of segments in each before-image recovery file belonging to TAF/CRM Data Manager. If you change this parameter, database recovery is not possible using existing before-image recovery files: you must recreate the before-image recovery files.
CMMBFL	50000B	Base field length in words for common memory manager CMM buffer management.
CMMEFL	0	Number of words for CMM to expand buffer management.
CRMUPM	15	Number of updates allowed. Also defines the number of records in each segment of the before-image recovery files.
BMAX	8	Number of before-image recovery files. The maximum value for BMAX is 63.
RMDM	1	Number of mainframes running TAF/CRM Data Manager.

The following parameters are used with TOTAL data manager and are defined in deck TAF. For information on the installation of TOTAL, refer to the NOS 2 Applications Installation Handbook.

Parameter	Released Default Value	Significance
TIMDM	10	Maximum number of tasks that can have outstanding requests to TOTAL at one time; additional tasks are put in recall.
TMAXDB	31	Maximum number of TOTAL data bases that can be initialized.
TMAXFIL	100	Maximum number of files per data base.

The following parameters are defined in deck COMKNWC.

Parameter	Released Default Value	Significance
MLIM	100	Maximum number of words in one SEND request before a task is rolled out pending completion of terminal output.
NCTL	250	Maximum number of terminals in network communication table (NCT). To reduce core storage requirements, NCTL may be less than the total number of terminals in the network file (each entry requires three CM words). NCTL should be greater than or equal to the maximum number of terminals logged in at one time. If NCTL is exceeded, a terminal is rejected upon login. If the number of terminals defined in the NCTFi file is less than NCTL, the number of terminals in NCTFi replaces the value specified by NCTL. NCTL is defined in COMKIPR.
NONTL	5*1000	When TAF begins execution at a control point, it attempts to establish communication with NAM. If the attempt fails or if the communication breaks down (for example, if NAM is not running), TAF continues attempting to establish communication every NONTL milliseconds.
WTIM	3*60	Maximum number of seconds that a task remains in the rollout queue if the task issues a SEND request with recall. The task is rolled in when WTIM seconds elapse or when TAF receives a supervisory message from NAM.

The following parameters specify the default communication block parameters.

Parameter	Released Default Value	Significance
CBDL	57	Length of the data input area in the communication blocks. This parameter is in deck COMKCBD.
CBUL	9	Length of user area in the communication blocks. This parameter is in deck COMKCBD.
NCBC	4	Maximum number of communication blocks reserved for large transaction input.
NLIN	4	Maximum number of users allowed to perform large transaction input simultaneously. TAF reserves n - NLIN x NCBC - RSCMB communication blocks for smaller transaction input. n is the number of communication blocks with which TAF is initialized. NLIN should not be less than 4.
RSCMB	2	Maximum reserved communication blocks for nonterminal use. This number is included in the NCMB parameter.

The following parameters, defined in deck COMKTLD, specify the default task library parameters.

Parameter	Released Default Value	Significance
TLDMT	600	Number of tasks per library. The maximum value for TLDMT is 1365.
TLDMN	10	Number of tasks that may be added on-line to TAF's copy of any particular task library directory by the LIBTASK TT option.
TRDMN	10	Number of transactions that may be added on-line to TAF's copy of any particular transaction library directory by the LIBTASK TT option.
TRDMT	300	Number of named transactions per library.

The following parameters, defined in deck DMREC (except where otherwise indicated), specify the default batch recovery parameters.

Parameter	Released Default Value	Significance
AAICL	200	Number of ignore entries.
CRMARB	15	Number of after-image records that will be buffered in the CM buffer for the file before they are flushed to disk. Also, the block length for after-image recovery files (ARFs). If you change this parameter, you must dump and recreate all ARFs. This parameter is in deck COMKIPR.
CRMARFN	35000	Length in physical record units (PRUs) of after-image recovery files. When preallocated by TAF or DMREC, the length specified by CRMARFN is assigned to the files excluding the header. This parameter is in deck COMKIPR.
DTTP	1	Tape drive type definition for dumping database and after-image recovery files; defined in deck COMKIPR. Value — Description 0 — Seven-track tapes. 1 — Nine-track tapes.
EXPCT	10	Default value of the percentage parameter for the EXPAND directive of deck DMREC.
FTABL	500	Number of concurrent active tasks.
NCOPY	2	Number of backup dumps to keep.

Parameter	Released Default Value	Significance
NDUMP	100	Number of dumps or directives. NDUMP must be less than 500_8.
NUMARF	1	Number of duplicate ARF copies.
TDEN	0	Tape density for dumps; any of the following. **Value** / **Description** 0 System default density 1 556 cpi 2 200 cpi 3 800 cpi 4 1600 cpi 5 6250 cpi This parameter is in deck COMKIPR.
TDTR	40+10*DTTP+TDEN	Tape format definition.
TTIGL	200	Number of transaction or task entries.
TLOGL	100	Number of files in data base.
TVSNL	40	Number of VSNs allowed.
WBUFL	4001B	Length in words of buffer used to contain data read from a block on the after-image recovery file. Size depends on the installation parameters CRMARB and CMDM, and on the maximum record length specified for the data base files in the xxJ file (refer to the TAF Reference Manual).

TEXT AND TEXTIO — PRODUCT TEXTS AND PRODUCT TEXTS I/O

General installation parameters related to the common products are defined within the common deck IPARAMS, included in Product Texts.

The default values of the IPARAMS configuration parameters are defined with the CEQU or CMICRO macros so that you can insert all modifications at one place. The CEQU and CMICRO macros define symbols conditionally; that is, they are effective only if the variables have not been previously defined. Therefore, any modifications you make must precede them. Insert all changes to IPARAMS at IPARAMS.15.

Modifications to be applied to products TEXT and TEXTIO should be applied only in the procedure TEXT.

To obtain a listing of all installation parameters in IPARAMS, run a job similar to the following:

```
job command.
USER,username,password,familyname.
CHARGE,*.
LABEL,TAPE,F=I,NT,D=PE,VSN=REL3A.
SKIPF,TAPE,12.
COPYBF,TAPE,OLDPL.
UPDATE,Q.
COMPASS,A,I,S=0.
--eor--
*COMPILE IPTEXT
--eoi--
```

The following list constitutes the extent of installation-changeable symbols in IPARAMS.

Parameter	Released Default Value	Significance
IP.CMU	0	If value is not zero, the compare/move unit hardware is present; if value is zero, the compare/move unit hardware is not present. The following common products reference IP.CMU. BAM 1 COBOL 5 Sort/Merge 4
IP.CSET	IP.C64.1	Defines the character set to be used throughout the system. The character set selected determines the collating sequence to be used; that is, the order in which records will be retrieved from a data base and the results of comparisons of characters on a basis of greater than or less than. Refer to the CYBER Record Manager Basic Access Methods Reference Manual for a description of the collating sequences. IP.C64.1 selects the CDC graphic 64-character set. To select the ASCII graphic 64-character set, specify a value of IP.C64.2 for the IP.CSET parameter.

Parameter	Released Default Value	Significance
		To select the CDC graphic 63-character set, specify the following two parameter definitions. IP.C63 EQU IP.C64.1 IP.CSET EQU IP.C63 To select the ASCII graphic 63-character set, specify the following two parameter definitions. IP.C63 EQU IP.C64.2 IP.CSET EQU IP.C63 The following common products reference IP.CSET. AAM 2; ALGOL-60 5; APL 2; BASIC 3; COBOL 5; COMPASS 3; FCL 4; FCL 5; FORTRAN 5; Sort/Merge 4; Query Update 3; Update 1; 8-Bit Subroutines 1
MODEL	74	Micro whose value is the model number of the CYBER 70 or CYBER 170 Computer System that corresponds to the type of central processor to be used for optimal code generation. Acceptable values are 71, 72, 73, 74, 171, 172, 173, 174, 175, 176, 720, 730, 740, 750, 760, 815, 825, 835, 855, 865, or 875. Most common products reference the MODEL micro.
IP.MECS SEE ERRATA	0	Maximum extended memory field length (in words) divided by 1000_8. IP.MECS is used by CYBER Loader to determine whether extended memory is available for use when loading user programs. IP.MECS is set to 7777_8 in the released binary on REL3A; if IPTEXT is assembled from the program library on REL3A, change IP.MECS (by placing appropriate Update directives on file USER) if you want extended memory available for use by CYBER Loader.
OS.ID	NOS 2.1	System identification micro for displaying the operating system name and version number in generated program binaries. Most common products reference the OS.ID micro.
IP.PD	6	Print density in lines per inch. Acceptable values are 6 and 8.
IP.PS	60D	Page size in lines. Maximum is 60 lines.

Parameter	Released Default Value	Significance

NOTE

The default settings for IP.PD and IP.PS work for both short (8 1/2 by 14 inch) and long (11 by 14 inch) paper. If you specify S (short) on the EST entry for the local batch printer (refer to Unit Record Equipment EST Entry in section 7), 60 lines per page will be printed; the print density parameter will be ignored. When using short paper, a page size specified at longer than 64 lines causes printing to continue over the page perforation. If users require 6-line-per-inch printing or a page size larger than the default, reserve one printer for long paper. Users can then route the print file with a special forms code to the correct printer.

Parameter	Released Default Value	Significance
HF.LIST	P74,S7	Micro whose value specifies the presence of certain hardware features in the configuration on which the products are being used. You should supply HF.LIST in addition to the MODEL micro, since use of various hardware features by the products is conditional on HF.LIST. However, if you do not define HF.LIST, the system selects a default value that is based on the MODEL micro and assumes no optional hardware. The default HF.LIST is a temporary capability that will be removed in a future release. You can define the following entries in HF.LIST.

Entry	Description
C	Compare/move unit (CMU) hardware is present.
L	For model 176, LCME is present. For models 815, 825, 835, 855, 865, and 865, UEM is present only if defined during deadstart. Both LCME and UEM are kinds of memory for which direct access instructions (014 and 015) are defined.

Parameter	Released Default Value	Significance

Entry	Description
Sn	Stack size; n specifies the size of the longest possible instruction stack program loop in words. If the mainframe being described has no stack, omit this entry. n can be either of the following.

n	Mainframe Model
7	74 and 6600
10	175, 176, 740, 750, 760, 865, and 875

Entry	Description
Px	Type of central processor; x can be one of the following values.

x	Model
S	6200, 6400, 6500, 71, 72, 73, 171, 172, 173, 174, 720, 730, 815, 825, 835, and 855; serial type CPU, etc.
74	6600, 6700, and 74.
175	175, 740, 750, and 760.
176	176.
740	740.
750	750.
760	760.
865	865.
875	875.

The processor type defaults to PS if HF.LIST is defined but the processor type is omitted.

Entry	Description
PSD	The central processor's exchange package contains a PSD register; model 176 only.
CRW	Central memory read/write operations are performed for 660/670 instructions; models 815, 825, 835, and 855, 865, and 875.

Parameter	Released Default Value	Significance

Entry — Description

Default values for HF.LIST are:

MODEL Micro Value	HF.LIST Default String
71	PS
72	C,PS
73	C,PS
74	P74,S7
171	PS
172	C,PS
173	C,PS
174	C,PS
175	P175,S10
176	P176,S10,L
720	C,PS
730	C,PS
740	P740,S10
750	P750,S10
760	P76,S10
825	PS,CRW,L
835	PS,CRW,L
855	PS,CRW,L
865	P865,S10,CRW,L
875	P875,S10,CRW,L
Any other	PS

Duplicate parameter entries (such as two Px entries) are not allowed.

When defining HF.LIST for products intended to be run on more than one mainframe, you can use the central processor type PS, P74, or P175 and include stack size (even if some of the mainframes do not have a stack). You must not include C and L unless those features exist on all of the mainframes in the configuration. The resulting products will not necessarily perform optimally on any one of the main-frames, but they will perform better on a parallel processor (such as a 175) if that processor type is set in HF.LIST.

TOOLS - MAINTENANCE TOOLS

No special information is needed to install Maintenance Tools.

TRACER - TRACER AND PROBE VERSION 1

No special information is needed to install Tracer and Probe.

UPDATE — COMMON MEMORY MANAGER VERSION 1, CYBER COMMON UTILITIES, AND UPDATE VERSION 1

The following Update features are available through assembly options. You can modify them by deleting the appropriate entry in the range UPDATE.703 through UPDATE.711. An attempt to use these features when the option is not assembled causes Update to issue error messages. For example, when PMODKEY is not set, the PULLMOD statement is not recognized as a legal directive.

Parameter	Released Default Value	Significance
DECLKEY	Enabled	Enables DECLARE directive.
CHAR64	Enabled	Declares 64-character set Update program library output.
PMODKEY	Enabled	Enables PULLMOD statement and G option.
AUDITKEY	Enabled	Allows audit functions.
EDITKEY	Enabled	Allows merge and edit functions.
OLDPLKEY	Enabled	Enables Update to read both old-style and new-style old program libraries.
EXTOVLP	Enabled	Enables detection of four types of overlap involving two or more cards in a correction set.
DYNAMFL	Enabled	Declares dynamic table expansion. When this option is assembled, Update automatically expands tables as required and dynamically requests NOS to change the user field length to accommodate the additional table area. At the end of the run, the field length is reduced to that requested by the user.

Update also uses the symbol IP.CSET and the MODEL micro defined in IPARAMS.

Common Memory Manager (CMM) uses symbol definitions from common deck CMMCOM. The symbols defined in IPTEXT that specify the operating system are also used. You can change the following CMMCOM installation parameters for CMM.

Parameter	Released Default Value	Significance
DEFVER	0	Defines which of two versions of CMM is to be used by default. Value — Parameter 0 — A version without error checking (FAST) is used. 1 — An error checking version (SAFE) is used.
FLF	2000B	When variable block code is not present (only fixed blocks exist), this value is used as a default by the field length reduction algorithm. The amount of free space above the highest fixed block is reduced to FLF central memory words.
FLINC	2000B	When field length is increased by CMM, this value is used as a default increase above the minimum amount needed.

XEDIT - XEDIT VERSION 3

Store the Extended Interactive Text Editor (XEDIT) help file, XEDITH, under user name LIBRARY. Use the MOVEPF utility, called by the following format of the DSD entry, to move XEDITH to the user index for LIBRARY. You must run MOVEPF at the system console.

```
X.MOVEPF(UI=index,DI=377776,F1=XEDITH)
```

Parameter	Description
index	User index under which XEDITH was stored during installation.

After moving XEDITH to user name LIBRARY, make it public with read permission.

INSTALLATION PARAMETERS IN NOS DECKS 6

You can modify installation parameters for the operating system by doing the following:

- Execute the COMBINE procedure, incorporating any corrective code (corrective code can change the deck line numbers). Refer to section 2.
- Get a listing of the deck that contains the parameter you want to change. From the listing, get information such as line numbers, which you need to change an installation parameter.
- Put the NOS installation deck parameter changes on file USER and again execute the COMBINE procedure.

If you change any of the installation parameters in a NOS deck, reassemble all routines that use that deck. Use the KRONREF command to determine which routines use the NOS deck.

Refer to table 6-1 for brief descriptions of the NOS common decks that contain installation parameters.

Table 6-1. NOS Common Decks (Sheet 1 of 2)

Common Deck Name	A Deck That Calls the Common Deck	Description
COMEIPR	CALLMSS	Mass Storage Subsystem parameters.
COMSACC	CALLSYS	User validation limits.
COMSBIO	CALLSYS	Central site batch I/O parameters.
COMSIOQ	CALLSYS	Dayfile/QPROTECT equivalences.
COMSJIO	CALLSYS	Devices to which users route files.
COMSLSD	CALLSYS	Search for label sector of a mass storage device.
COMSMSC	CALLSYS	Miscellaneous parameters for the operating system.
COMSMTX	CALLSYS	Magnetic tape executive routine and magnetic tape processing routine parameters.
COMSPFM	CALLSYS	Permanent file symbols and locations and formats of call blocks, catalog, and permit entries.
COMSPRO	CALLSYS	PROFILa parameters.
COMSREM	CALLSYS	Interactive Facility parameters.
COMSRSX	CALLSYS	Job resource executive parameters.

Table 6-1. NOS Common Decks (Sheet 2 of 2)

Common Deck Name	A Deck That Calls the Common Deck	Description
COMSSFS	CALLSYS	Field length limit for execution of MODVAL and PROFILE commands.
COMSSRU	CALLSYS	Parameters used in SRU calculations.
COMSSSJ	CALLSYS	Special system job parameters.
COMS1DS	CALLSYS	1DS function code definitions.
COMTNAP	CALLTAB	Valid network application parameters.
COMUSIT	CALLMSS	ASMOVE utility parameters.
PPCOM	PPTEXT	Extended memory space for system use, length of L-display input and output buffers, and number (plus 1) of mass storage devices.

To assemble a common deck, run a job similar to the following. This job assumes that RELSOPL was copied to a direct access file named OPL after the COMBINE procedure was executed. OPL is under the user name LIBRARY.

Job	Comment
job command.	
USER,username,password,familyname.	
CHARGE,*.	
ATTACH,OPL/UN=LIBRARY.	User must have permission to access OPL.
COPY,INPUT,IN.	
REWIND,IN.	
MODIFY,Z./*CREATE,IN/*EDIT,LISTDCK	
COMPASS,I,S=NOSTEXT.	
--eor--	
LISTDCK	
IDENT LISTDCK	
*CALL common deck name	common deck name is CALLMSS, CALLPPU, CALLSYS, CALLDIS, or CALLTAB.
END	
--eoi--	

To get a source listing of a common deck, run a job similar to the following. This job assumes that RELSOPL was copied to a direct access file named OPL after the COMBINE procedure was executed. OPL is under the user name LIBRARY.

Job	Comment
job command.	
USER,username,password,familyname.	
CHARGE,*.	
ATTACH,OPL/UN=LIBRARY.	User must have permission to access OPL.
MODIFY,LO=A,C=0,Z,L=LIST./*EDIT,deck	deck is the name of a common deck.
ROUTE,LIST,DC=PR.	
--eoi--	

The following deck is also discussed in this section. It is not a common deck.

Deck	Description
DSD	DSD command syntax macro call parameters.

To get a listing of DSD, run a job similar to the following (this job assumes that RELSOPL was copied to a direct access file named OPL after the COMBINE procedure was executed). OPL is under the user name LIBRARY.

Job	Comment
job command.	
USER,username,password,familyname.	
CHARGE,*.	
ATTACH,OPL/UN=LIBRARY.	User must have permission to access OPL.
MODIFY,Z./*EDIT,DSD	
COPYSBF,COMPILE.	
--eoi--	

To assemble DSD, run a job similar to the following. This job assumes RELSOPL is saved as a direct access file named OPL under the user name LIBRARY.

Job	Comment
job command.	
USER,username,password,familyname.	
CHARGE,*.	
ATTACH,OPL/UN=LIBRARY.	User must have permission to access OPL.
MODIFY,A,Z./*EDIT,DSD	
COMPASS,A,I,S=NOSTEXT,B=0.	
--eoi--	

COMEIPR PARAMETERS

COMEIPR contains the following parameters used by the Mass Storage Subsystem executive (MSSEXEC). Assemble CALLMSS to obtain a listing of COMEIPR.

Parameter	Released Default Value	Significance
NUMSLV	2	The number of slave mainframes for which the master MSSEXEC can service file staging requests.
NUMRB	9	The number of file staging request blocks available to a slave mainframe.
SLRP$INTV	5	The number of seconds that MSSEXEC waits to look for new staging requests from the slave mainframes.
SLAV$INTV	60	The number of seconds that MSSEXEC waits with no signal from a slave mainframe before assuming that the slave executive has terminated.

COMSACC PARAMETERS

COMSACC contains a general description of the user validation file. Assemble CALLSYS to obtain a listing of COMSACC.

Parameter	Released Default Value	Significance
APFN	VALIDUZ	Micro definition that specifies the name of the file containing the user names that validate user access to the operating system. Refer to the NOS 2 System Maintenance Reference Manual for further information on VALIDUs.
AUFN	VALINDZ	Micro definition that specifies the name of the available user indexes file. Refer to the NOS 2 System Maintenance Reference Manual for further information on VALINDs.

The NOS 2 System Maintenance Reference Manual describes the use of the following COMSACC user control parameters.

Parameter	Released Default Value	Significance
KTLI	10B	Default time limit; the maximum value is 176_8.
KLPI	1000B	Default limit for lines printed from a file; the maximum value is 3776_8.
KCPI	0	Default limit for cards punched from a file; the maximum value is 76_8.
KMSI	1000B	Default limit for additionally allocated mass storage PRUs; the maximum value is 7776_8.
KDFI	100B	Default limit for dayfile messages written; the maximum value is 176_8.
KCCI	100B	Default limit for commands processed; the maximum value is 176_8.
KECI	0	Default limit for extended memory field length/1000_8; the maximum value is 176_8.
KCMI	37B	Default limit for central memory field length/100_8; the maximum value is 37_8.
KSLI	10B	Default limit for SRU accumulation; the maximum value is 76_8.
KDTI	0	Default limit for the number of detached jobs; the maximum value is 37_8.
KPTI	1000B	Default limit for the number of units plotted; the maximum value is 76000_8.

COMSBIO PARAMETERS

COMSBIO contains the following parameters, which are used for control of BIO functions. Assemble CALLSYS to obtain a listing of COMSBIO.

Parameter	Released Default Value	Significance
PL6L	64	Number of lines of print a user is charged for each page of output printed by BIO at six lines per inch.
PL8L	85	Number of lines of print a user is charged for each page of output printed by BIO at eight lines per inch.

COMSIOQ PARAMETERS

COMSIOQ contains the following parameter, which is used for control of terminated dayfiles. Assemble CALLSYS to obtain a listing of COMSIOQ.

Parameter	Released Default Value	Significance
USRN	null	Micro definition that specifies the user name to which terminated dayfiles should be permitted.

COMSJIO PARAMETERS

COMSJIO contains the following parameters, which define the devices to which the site allows users to route files. Two-character disposition codes, corresponding to the device codes defined for the ROUTE command, followed by a $ identify the legal devices. Assemble CALLSYS to obtain a listing of COMSJIO.

Parameter	Released Default Value	Significance
LP$	Defined	Any line printer.
PR$	Defined	Any line printer.
LR$	Defined	Any 580-12 printer.
LS$	Defined	Any 580-16 printer.
LT$	Defined	Any 580-20 printer.
SB$	Defined	Punch system binary.
PB$	Defined	Punch system binary.
P8$	Defined	Punch 80-column binary.
PU$	Defined	Punch coded.
PH$	Defined	Punch coded.
PL$	Defined	Plotter.

COMSLSD PARAMETERS

COMSLSD contains the following parameters, which reference information maintained in the label sector of a mass storage device. Assemble CALLSYS to obtain a listing of COMSLSD.

Parameter	Released Default Value	Significance
LTKL	20B	If you did not initialize a mass storage device during deadstart (using the INITIALIZE entry described in the NOS 2 System Maintenance Reference Manual), the system searches the device for a label that might be in track 0. This parameter specifies the number of tracks the system will search before determining that the device has a bad label or no label. When it reaches that track number (in the released system, track 20_8), it stops searching for a label. If the device is a system device, the system writes a new label; if it is not a system device, the error codes LE (label error) and U (unavailable) status are entered in the mass storage table (MST), and the device must be initialized after deadstart. MST is described in the NOS 2 Operator/Analyst Handbook.

COMSMSC PARAMETERS

COMSMSC contains the following miscellaneous parameters, which is used by the operating system. Assemble CALLSYS to obtain a listing of COMSMSC.

Parameter	Released Default Value	Significance
MXSY	5	Maximum number of devices that can be defined as system devices during deadstart (refer to SYSTEM - System Library Device Assignment in section 7).
AFDL	20000B	Account file threshold size in PRUs.†
BLTL	1000B	Binary maintenance log threshold size in PRUs. †
DFDL	20000B	Dayfile threshold size in PRUs.†
ELDL	1000B	Error log threshold size in PRUs.†

†When entries in any of these files reach the threshold, the A,OPERATOR flashing message appears on the console screen (refer to the NOS 2 Operator/Analyst Handbook).

COMSMTX PARAMETERS

COMSMTX contains the following parameters, which are used by the magnetic tape executive routine and by related magnetic tape processing routines. Assemble CALLSYS to obtain a listing of COMSMTX.

Parameter	Released Default Value	Significance
MUNIT	16D	Maximum number of tape units defined per mainframe.
POLM	0	Flag indicating whether all tape hardware error messages will be issued to the user job dayfile. If POLM is 0, the system issues only the first and last messages to the user dayfile. If POLM is 1, the system issues all tape error messages to the user dayfile. The user can override the installation setting of POLM with parameters on the tape assignment command (refer to the NOS 2 Reference Set, Volume 3). The system issues all tape error messages to the error log regardless of the setting of POLM.
POGH	0	Flag indicating whether the system allows hardware-detected correctable errors when writing on 6250-cpi group-encoded (GE) tapes. The user can override the installation setting of POGH with parameters on the tape assignment command (refer to the NOS 2 Reference Set, Volume 3). If POGH is 0, the tape subsystem performs write error correction according to industry standard group-coded recording (GCR) techniques. Control Data recommends this setting because it provides efficient throughput, error recovery, and tape use when writing GE tapes on media suitable for use at 1600 cpi or 6250 cpi. If POGH is 1, hardware GCR error correction is disabled. Control Data recommends this option only for special archiving and diagnostic applications. Successful use requires higher-than-normal quality tape and special drive adjustments. Use in a normal environment generally results in increased error rates, decreased throughput, and decreased tape capacity. Use only tape that is suitable for recording at 6250 cpi, when this setting of POGH is in effect. Because use of the disabled GCR error correction mode (also known as perfect write) may necessitate additional maintenance activities, consult site maintenance personnel before making this the default mode of operation.
ZFAM	A null micro	Enables conversion of binary zero family names to nonzero family names. ZFAM allows users to continue to access labeled tapes that are restricted to owner access (file accessibility field in HDR1 label is A) and were built under the binary zero family. If ZFAM is a null micro, the system default family name is substituted for the binary zero family name; otherwise, ZFAM specifies the name to be substituted.

COMSPFM PARAMETERS

COMSPFM contains the following parameters, which are used for permanent file symbols and locations, formats of call blocks, and catalog and permit entries. Assemble CALLSYS to obtain a listing of COMSPFM.

Parameter	Released Default Value	Significance
APLO	0	Auxiliary pack load option. This parameter controls whether or not a user can request an auxiliary pack to be loaded via an MFLINK request. When APLO equals 0, MFLINK requests to auxiliary packs not currently mounted will be rejected with the message DEVICE UNAVAILABLE. When APLO is equated to a nonzero value, such MFLINK requests may roll out while waiting for the pack to be mounted (provided the user specified the NA or WB parameter). Since there is no global resource executive in an LCN environment, a potential of deadlock exists in the latter instance.
DFPT	DI1	Equipment type. When accessing an auxiliary device with a permanent file command, the permanent file manager checks that the equipment type and pack name of the device match the equipment type (R parameter) and pack name on the command. If R is not specified, the system uses the equipment type specified by DFPT. If the default is used for another equipment type, the error message ILLEGAL DEVICE REQUEST occurs.
MNHS	5	Minimum size hole, in sectors, that permanent file manager (PFM) creates in the indirect access file chain when using an existing hole. If, in the search for a hole in which to save an indirect access file, PFM finds that the use of an existing hole creates a new hole containing fewer sectors than MNHS, then PFM allocates space at the end of the indirect access chain. If a delink operation creates a hole smaller than MNHS, PFM delinks one less track to ensure minimum size for the hole. The purging of a file whose total length is less than MNHS results in the creation of a hole smaller than MNHS. If a value for MNHS is smaller than the average length of the indirect access files on the system, it results in holes that may be unusable. If the value is larger than the average file length, it results in holes which are not used for a period of time. For efficient use of holes, the value for MNHS should be close to the average length of the indirect access files on the system.

Parameter	Released Default Value	Significance
BRDE	BRAL	Backup requirement (BR) default specifications; can be set to the following symbolic values.

Value	Description
BRAL	Backup always required.
BRMD	Media-dependent backup for systems with an alternative mass storage facility.
BRNO	No backup required.

Parameter	Released Default Value	Significance
RSDE	RSNP	Preferred residence (PR) default specification; can be set to the following symbolic values.

Value	Description
RSMS	Mass storage facility residence preferred.
RSNP	No preferred residence.

For individual users, each of four permanent file access limits is established through MODVAL (refer to the NOS 2 System Maintenance Reference Manual) by specifying a range index from 0 through 7. Each range index corresponds to an upper limit specified by one of the following installation parameters. The last character of the installation parameter indicates the range index being defined. Table 6-2 summarizes the released values for each parameter. Setting a parameter to 0 indicates unlimited access.

Parameter	Significance
NFRNGn	Upper limit of range n for file count; must not exceed 77777_8.
CSRNGn	Upper limit of range n for cumulative size of indirect access files, specified in PRUs; must not exceed 777777_8.
FSRNGn	Upper limit of range n for size of individual indirect access files, specified in PRUs; must not exceed 77777_8.
DSRNGn	Upper limit of range n for size of individual direct access files, specified in PRUs; must not exceed 777777_8.

Table 6-2. Released Values of Permanent File Limit Ranges

Parameter	Values of n†						
	1	2	3	4	5	6	7
NFRNGn	10	20	30	40	50	100	0
CSRNGn	1000	2000	5000	10000	50000	100000	0
FSRNGn	10	30	50	100	150	300	0
DSRNGn	1000	2000	5000	10000	50000	100000	0

†All values are specified in octal; 0 indicates unlimited access.

COMSPRO PARAMETERS

The following COMSPRO parameters contain a general description of the PROFILa file. Assemble CALLSYS to obtain a listing of COMSPRO.

Parameter	Released Default Value	Significance
PPFN	PROFILB	Micro definition specifying the PROFILE routine's database file name (refer to the NOS 2 System Maintenance Reference Manual).
PPWD	SECURUS	Micro definition specifying the PROFILE routine's database file password.
PUSN	SYSTEMX	Micro definition specifying the catalog location of the PROFILE routine's database.

COMSREM PARAMETERS

COMSREM contains the following parameters, which are used by the Interactive Facility executive. Assemble CALLSYS to obtain a listing of COMSREM.

Parameter	Released Default Value	Significance
UTIS	100B	Default CPU time limit in seconds for any particular terminal job's activity, if it is not specified with the SETTL command (refer to the NOS 2 Reference Set, Volume 3).
VDSI VDTI	100B 100B	Default system resource unit (SRU) and time limit increment values for the S,nnnnn and T,nnnnn interactive commands.
VXLL	160D	Maximum number of characters in a logical input line.
VXPH	160D	Determines the physical line length that IAF accepts. IAF uses VXPH to calculate a buffer length.

COMSRSX PARAMETERS

COMSRSX contains the following parameters, which are used by the resource executive. Assemble CALLSYS to obtain a listing of COMSRSX.

Parameter	Released Default Value in Minutes	Significance
RPMS	4	Length of time that a job waiting for an auxiliary device is rolled out before retrying assignment.
RPOV	8	Length of time that a job that has had a request for an auxiliary device denied due to overcommitment deadlocks is rolled out before retrying assignment.
SUBM	10	If MAGNET is not active, length of time that a noninteractive service class job calling RESEX is rolled out before retrying assignment.
MTMS	2	When one of the following situations prevents immediate assignment of the tape, MTMS specifies the length of time that a job requesting a tape is rolled out before retrying the assignment. ● The job requests a tape with a VSN that is not currently available. ● The job requests a nine-track tape that is mounted without a write ring, the job requests the wrong tape density, and the tape hardware detects that the density of the tape is incompatible with the unit on which it is mounted (800-cpi tape on a 1600/6250-cpi drive or 6250-cpi tape on an 800/1600-cpi drive).
RFTL	10	Length of time that a job requesting a resource is rolled out before retrying the request when a track limit occurs on the resource demand or VSN files.
MTOV	8	Length of time that a job that has had a request for a magnetic tape denied due to overcommitment deadlocks is rolled out before retrying the assignment.

COMSSFS PARAMETERS

COMSSFS parameters are used by the MODVAL and PROFILE commands. Assemble CALLSYS to get a listing of COMSSFS.

Parameter	Released Default Value	Significance
FLLM	50000B	Specifies the field length limit for the execution of the MODVAL and PROFILE commands. If the execution of a MODVAL or PROFILE command requires more than the specified field length, disk storage is used. Accessing disk storage is more time consuming than accessing central memory and will degrade performance.

COMSSRU PARAMETERS

COMSSRU contains the parameters used in SRU calculations. Assemble CALLSYS to obtain a listing of COMSSRU. Refer to COMSSRU in the NOS 2 System Maintenance Reference Manual and to CPM in section 9.

COMSSSJ PARAMETERS

COMSSSJ contains the following parameters, which are used by special system jobs. Assemble CALLSYS to obtain a listing of COMSSSJ.

Parameter	Released Default Value	Significance
ART	4 minutes	Length of time that a job is rolled out while waiting for a direct access file to become available before trying to access it again. This value specifies the default for the WB parameter on the ATTACH command.
FRT	15 seconds	Length of time that a special system job is rolled out when a fast attach file is busy.

COMS1DS PARAMETERS

COMS1DS contains the following parameter used for DSD jobs. Assemble CALLSYS to obtain a listing of COMS1DS.

Parameter	Released Default Value	Significance
SYSCHG	SYSTEM	Micro definition that specifies the system charge number for jobs initiated by DSD.

COMTNAP PARAMETERS

COMTNAP defines a table that maps valid network application names to the bit position of the access word that must be set to allow use of one or more network applications (refer to the description of MODVAL in the NOS 2 System Maintenance Reference Manual). Assemble CALLTAB to obtain a listing of COMTNAP. This common deck does not contain any executable code. Each table entry has the following format.

59	17	11 0
application name (6-bit display code, left-justified, blank-filled)	reserved	access word bit position

Bits 17 through 12 of each entry are reserved for the program that uses COMTNAP. These bits are set to zero when COMTNAP is assembled. The last word of the table must be zero.

Each application defined in COMTNAP must appear only once. However, any access word application bit can appear more than once; that is, a given access word application bit can be defined to permit use of more than one application, if the operations at a particular site make such a definition desirable. Bits 47 through 36 of the access word are reserved for customer application use; bits 35 through 24 are reserved for Control Data application use.

The released table defined by COMTNAP is:

Word	Contents 59	17	11 0
TNAV	IAF	0	24
	RBF	0	25
	TAF	0	26
	MCS	0	27
	TVF	0	28
	reserved	0	29
	reserved	0	30
	CS	0	31
	APP1	0	32
	APP2	0	33
	APP3	0	34
	PLATO	0	35
TNAV+TNAVL	0		

The following table-related parameters are defined in COMTNAP. All symbols are unqualified.

Parameter	Released Default Value	Significance
TNAV	-	Table first word address. Program that uses COMTNAP defines the value.
TNAVL	12	Table length, excluding zero-word terminator.

COMUSIT PARAMETERS

COMUSIT contains the following parameters, which are used by the ASMOVE utility to control selection of files for destaging to the mass storage facility (MSF) and for releasing their disk space. Compile CALLMSS to obtain a listing of COMUSIT. Compile ASMOVE to determine how the following weight and scale factors are used in the selection algorithm.

Parameter	Released Default Value	Significance
DEFDB	2	Weight factor for MSF-preferred residence.
DB$SCALE	1.0	Scale factor for MSF-preferred residence.
DEFDC	1	Weight factor for no preferred residence.
DC$SCALE	1.0	Scale factor for no preferred residence.
DEFDL	1	Weight factor for file length.
DL$SCALE	1.0	Scale factor for file length.
DEFDT	0	Weight factor for time since last modification.
DT$SCALE	1.0	Scale factor for time since last modification.
DEFDV	24	Weight factor for destage control value.
DV$SCALE	25.0	Scale factor for destage control value.
DEFMN	6	Weight factor for minimum file length.
MN$SCALE	25	Scale factor for minimum file length.
DEFMX	128	Weight factor for maximum file length.
MX$SCALE	250	Scale factor for maximum file length.

PPCOM PARAMETERS

PPCOM contains the following parameters, which are used by system peripheral processor packages for intercommunication. Assemble NOSTEXT to obtain a listing of PPCOM.

Parameter	Released Default Value	Significance
ECSY	0	Number of words/1000_8 of user extended memory space reserved for system use.
LCOM	12	Maximum length of the L-display input buffer in words. The value for LCOM can range from 1 through 12_8.
LDSY	350	Maximum length of the L-display output buffer in words. The value for LDSY can range from 100_8 through 1000_8.
NMSD	40	Highest EST ordinal plus 1 that can define mass storage devices. Therefore, in the released system, ordinals 1 through 37_8 can define mass storage devices. The maximum value for NMSD is 40_8.

The assembly constant INSP$ is defined in NOSTEXT. If the INSP$ reference is deleted, 10 bytes in both the DSD display and the command overlays are unavailable for site code.

DSD PARAMETERS

The following parameters, specified in ENTER macro calls (within the DSD syntax tables), cause the first 25 characters of the associated DSD command to be logged in the system dayfile and/or the error log. The commands are logged just as they are entered by the operator except that the characters

DS,

are placed before each command. The DSD listing contains an explanation of the ENTER macro. Assemble DSD to obtain a listing.

Parameter	Released Default Value	Significance
SDF	-	When specified in an ENTER macro call, the associated command is logged in the system dayfile.
ERL	-	When specified in an ENTER macro call, the associated command is logged in the error log. On the release tapes, the OFF, ON, channel control, and memory commands specify ERL on their ENTER macro calls.

CMRDECK 7

The central memory resident deck (CMRDECK) resides on the deadstart tape as a text record which is processed during system initialization. It contains entries defining the following types of information.

- Central memory.
- Nonmass storage equipment.
- Mass storage equipment.

In this section, a CMRDECK entry is described under the type of information it defines. An alphabetical listing of the CMRDECK entries with page numbers is contained on the inside front cover.

The deadstart tape can contain up to 100_8 CMRDECKs. Having several CMRDECKs on the same deadstart tape is advantageous because one tape can deadstart several configurations. You can obtain of listing of all CMRDECKs by accessing the SYSTEM file with an ASSIGN or COMMON command, then using the T parameter on the CATALOG command (refer to the NOS 2 Reference Set, Volume 3 for more information concerning these commands).

You can modify the released settings of the CMRDECK in two ways: either by typing a new entry when the CMRDECK is displayed during deadstart or by creating a new deadstart tape. The usual method of modifying a CMRDECK is:

1. Deadstart, using the released deadstart tape, and select the CMRINST and the CMRDECK to be displayed (refer to the NOS 2 Operator/Analyst Handbook).

   CMRINST lists all valid CMRDECK entries (the released version of CMRINST is shown in appendix D). Default values, described in this section, are assumed if the entries do not appear in the CMRDECK being used. If either CMRDECK or CMRINST overflows two screens, you can advance the display by pressing the + key.

2. Modify the released version of CMRDECK by entering the changes or additions from the system console while either the CMRDECK or CMRINST is displayed. Each console entry supersedes the value currently specified in the CMRDECK (or the default value).

   NOTE

   The modified CMRDECK remains in effect only until the next deadstart is performed, except for a level 3 deadstart. That is, changes to the CMRDECK are not recovered across level 0, 1, and 2 deadstarts unless a new deadstart tape is created to reflect them.

3. To expedite subsequent deadstarts, modify the CMRDECK on the deadstart tape using GENSYS (refer to Step 3 - Create a Deadstart Tape in section 2).

When constructing or modifying a CMRDECK, the following restrictions apply.

- The equipment assignment entry (EQ) must precede any other assignments for a device (such as assigning it for permanent file, system, or temporary file use). If you modify the EQ entry, reenter all other assignments for that equipment.
- The device from which you are deadstarting must be defined.
- Commas must separate parameters.
- All parameters are order dependent and must be specified unless they are optional. Be sure to include commas to note omitted parameters.
- A period must terminate each entry.
- Except where explicitly specified that controllers and/or equipment can be shared between mainframes, assume they cannot be shared. For example, NOS does not support sharing a two-channel tape controller between mainframes; nor does it support sharing mass storage controllers except as specified in the multimainframe sections.

An arrow (↑) points to an error in an entry typed at the system console. When an error exists in an entry in a CMRDECK on the deadstart tape, the CMRDECK is displayed with an arrow indicating the error. This occurs even if you do not select the display CMRDECK option.

CENTRAL MEMORY DESCRIPTIONS

The general function of central memory description entries is to assign the amount of central memory to be used for central memory resident (CMR) and the amount to be used for job processing. The simplified relationship is: the more central memory that is assigned to dayfile buffers in CMR, the less that is available for job field lengths.

If, for example, you need a large portion of central memory to run a job, it might be advisable to decrease the size of the dayfile buffers area in CMR in order to accommodate that job. However, when the dayfile buffers are smaller, the information stored in them is written to mass storage more often, which requires more system overhead.

If you will run only a few batch jobs, fewer control points may be required. Thus, you could decrease the control point area in CMR (which requires 200_8 words per control point).

The following entries are specified in the SET program with the released default values indicated.

Entry Format	Released Default Value	Significance
ACCOUNT=eq,length.	0,400_8	Sets the residence of the account dayfile and the length of the account dayfile buffer. The account dayfile is an accounting record containing such information as kind and amount of resources used and jobs and execution times. This account information is written to the central memory account dayfile buffer during job processing; the central memory buffer is written to mass storage when it is full. The account dayfile buffer resides in CMR in the dayfile buffer area.

Entry Format	Released Default Value	Significance
		Parameter / **Description** eq — One- or two-digit octal equipment status table (EST) ordinal of the equipment on which the account dayfile is to reside. A value of zero means the system uses the first nonremovable device it can find. If the existing account dayfile is recovered, the account dayfile buffer resides on that equipment and the eq parameter is ignored. length — Three- or four-digit octal length of the account dayfile buffer in CMR; must be a multiple of 100_8. If 0 is specified, messages issued to the account dayfile are discarded.
DAYFILE=eq,length.	$0,400_8$	Sets the residence of the system dayfile and the length of the system dayfile buffer. The dayfile buffer contains the dayfile information, which is maintained in the same way as the account dayfile buffer. It resides in CMR in the dayfile buffer area. **Parameter** / **Description** eq — One- or two-digit octal EST ordinal of equipment on which the dayfile is to reside. A value of zero means the system uses the first nonremovable device it can find. The residence of this dayfile is normally determined by the recovery of the existing dayfile. Use this parameter if no system dayfiles are recovered. length — Three- or four-digit octal length of the system dayfile buffer in CMR; must be a multiple of 100_8. If 0 is specified, messages issued to the system dayfile are discarded.
EJT=number.	620_8	Specifies an octal number of entries for the executing job table (EJT). The system uses the EJT entries to keep track of executing jobs. The maximum value for number is 7777_8; the minimum value is 3.

Entry Format	Released Default Value	Significance
ERRLOG=eq,length.	$0,100_8$	Sets the residence of the error log dayfile and the length of the error log dayfile buffer. The error log dayfile is a record of error messages and the execution time for a particular job. This information is maintained in the same way as the account dayfile buffer. **Parameter** / **Description** eq — One- or two-digit octal EST ordinal of equipment on which the error log dayfile is to reside. A value of zero means the system uses the first nonremovable device it can find. If the existing error log dayfile is recovered, the error log dayfile buffer resides on that equipment and the eq parameter is ignored. length — Three- or four-digit octal length of the error log dayfile buffer in CMR; must be a multiple of 100_8. If 0 is specified, messages issued to the error log dayfile are discarded.
FNT=number.	23_8	Sets the octal number of entries allowed in the system file name table (FNT). The system FNT contains the system file and all fast-attach files. Determine the necessary number of FNT entries by the formula: number = SY + RS + (VL + PR) x FM + SFA **Variable** / **Description** SY — Number of system files. RS — Number of resource files. VL — Number of VALIDUs files per family. PR — Number of PROFILa files per family. FM — Number of families that can be active at any one time. SFA — Site defined fast-attach files.

Entry Format	Released Default Value	Significance
		For a system installed with the released defaults and with no additional files added to the system FNT by local code, the maximum number of FNT entries necessary is 201_8. The maximum value for number is 7777_8; the minimum value is 3.
FOT=number.	10_8	Sets the number of entries in the family ordinal table (FOT). Each family is allowed one entry in the table. The first entry in the FOT is reserved for system use. The system uses family name and user index for job ownership and file routing. The family ordinal is a 6-bit number that corresponds to a particular family name. The FOT maintains the family name to family ordinal relationship. If a family is unloaded and later reloaded, it continues to use the same family ordinal. The maximum value for number is 100_8; the minimum value is 3. The size of the FOT need not be the same for each mainframe in a multimainframe environment.
IPD=iprdeck.	First IPRDECK on deadstart file	Specifies the name of the IPRDECK to use at deadstart. The IPRDECK contains installation parameters (defined in section 9). Up through 64 IPRDECKs can exist on a deadstart file. If an IPD entry is not included in the CMRDECK, the first IPRDECK on the deadstart tape is processed and is not displayed.
LIB=n.	0	Indicates which LIBDECK to use in building the system. Up through eight LIBDECKs can exist on a deadstart file. n can be from 0 through 7. If 0 is specified, the record named LIBDECK is used; if 1 is specified, LIBDCK1 is used; and so forth. LIBDECK is a directive record used by SYSEDIT (refer to section 10).
LIDT=length	10_8	Specifies the number of logical identifiers (LIDs) in the LID table. The LID table contains the LIDs specified in network configuration statements. The maximum value is 777_8; the minimum value is zero.

Entry Format	Released Default Value	Significance
MAINLOG=eq,length	~~0,0~~ 0,1001B	Sets the residence of the binary maintenance log and the length of the binary maintenance log buffer. The binary maintenance log is a record of hardware diagnostic information.

Parameter	Description
eq	One- or two-digit octal EST ordinal of equipment on which the binary maintenance log is to reside. A value of zero means the system uses the first nonremovable device it can find. If the existing binary maintenance log is recovered, the binary maintenance log buffer resides on that equipment and the eq parameter is ignored.
length	Three- or four-digit octal length of the binary maintenance log buffer in CMR; must be a multiple of 100_8. If ~~length~~=0, messages issued to the binary maintenance log are discarded.

Entry Format	Released Default Value	Significance
MID=id.	AA	Specifies the two-character machine identification (id) which is associated with the mainframe. The id characters must be alphanumeric.
MINCM=size.	1400_8	Reserves an amount of central memory for system operation (UEM is defined in the extended memory equipment EST entry). The minimum and default value for size is 49K. If you specify a value below 32K, the system may not deadstart properly.

Entry Format	Released Default Value	Significance

Parameter	Description
size	One- to four-digit octal value (one- to five-digit octal value for a model 815, 825, 835, 855, 865, or 875) that restricts the actual central memory size; this value represents the amount of central memory in hundreds (octal); value cannot be 0 and cannot exceed the total number of words of memory present in the machine.

Memory	Size
49K	1400
65K	2000
98K	3000
131K	4000
198K	6000
262K	10000

Entry Format	Released Default Value	Significance
NAME=date line.	CDC NETWORK OPERATING SYSTEM	Specifies the system date line that is displayed on the system console display and on the terminal, when an interactive user logs in to the system.

Parameter	Description
date line	Alphanumeric-character system date line; must be less than 39 characters in length.

Entry Format	Released Default Value	Significance
NAMIAF=maxt,hs.	200_8,0	Specifies the number of network terminals and the number of high-speed terminals (character speed greater than 120 cps) that can be connected to IAF at one time.

Parameter	Description
maxt	Total number of network terminals; can be from 1 through 1440_8, when there are no high-speed terminals. If there are high-speed terminals, you must calculate maxt. Do the following calculation in decimal, round maxt down to an integer value, and convert it to an octal value for the NAMIAF entry. Convert hs to a decimal number before multiplying it by 2.4. maxt = 800 - 2.4hs
hs	Total octal number of high-speed terminals.

Entry Format	Released Default Value	Significance
NCP=number.	12_8	Sets the number of control points available for job processing. Refer to Job Control in section 9 for a discussion of the proper number of control points to select. **Parameter** — **Description** number — Number of control points available in central memory; number can be from 2 through 33_8.
PPU=pp_1,pp_2,...,pp_n.	All available PPs are active	Toggles the active status of the physically available PPs,† except for PPs 1, 2, and 10, which are always active. Active means the PP is available for system use; inactive means it is not available for system use. The PPU entry is not in the released CMRDECK, therefore all available PPs are active. PPU is a toggle entry: each entry changes the active status of the PPs. **Parameter** — **Description** pp_i — Number of the PP; from 0 through 11_8 and 20_8 through 31_8. Specifying an asterisk (*) on the entry instead of pp_i toggles the status of PPs 20_8 through 31_8. This entry may be useful if PP memory is failing or if a channel is causing problems on its associated PP. For example, the following deactivates PP 3 and PP 4 (assuming no other entries have been made for PP 3 and PP 4). PPU=3,4.
QFT=number.	620_8	Sets the number of entries allowed for the queued file table (QFT). The system uses the QFT to manage all files in the input and output queues. number is a one- through four-digit number from 3 through 7777_8.
VERSION=name	NOS 2.1-580/577	Specifies the system version that is displayed on the system console display. **Parameter** — **Description** name — Alphanumeric-character version name; must be less than 19 characters in length.

† A PP that has been turned off by CTI is physically unavailable and cannot be turned on by the PPU entry.

EQUIPMENT ASSIGNMENTS: NONMASS STORAGE

The following EST entries are described in this subsection.

Clear EST assignment
Nonstandard equipment
Dummy equipment
System console display equipment
Unit record equipment
Magnetic tape equipment
Mass storage facility equipment
Stimulator equipment
Network processing unit
Network Access Device
Two-port Multiplexer
MAP III equipment

CLEAR EST ASSIGNMENT ENTRY

Use the following entry to clear an assignment that currently exists for an EST ordinal. Clearing the assignment does not clear flaw entries for that equipment.

```
EQeq=0.
  or
EQeq=.
```

Parameter	Description
eq	EST ordinal of the equipment; eq can be from 1 through 74_8.

NONSTANDARD EQUIPMENT EST ENTRY

The nonstandard equipment EST entry allows you to define nonstandard equipment or to add site debugging modifications.

Use the following format to enter the actual octal value that is to reside at that EST ordinal.

```
EQeq=value.
```

Parameter	Description
eq	EST ordinal of the equipment; eq can be from 1 through 74_8.
value	From 1- through 20-digit octal value; this value is entered in the EST word for the specified ordinal. The word is right-justified and zero-filled if value has fewer than 20 digits.

PSEUDO EQUIPMENT EST ENTRIES

The system automatically reserves EST ordinals 0, 75_8, 76_8, and 77_8 for pseudo equipment EST entries: they cannot be used for other equipment definitions. You can reenter the pseudo equipment entries (RD† equipment for ordinal 0, TT† equipment for ordinal 75_8, TE equipment for ordinal 76_8, and NE† equipment for ordinal 77_8) on the CMRDECK either from the deadstart tape or as a console entry. You cannot change the equipment type nor can you remove or declare pseudo equipment; however, you can specify channels and other parameters. The system uses some of the fields in EST entries for pseudo equipment, and you can use others to contain configuration-dependent information for stimulators and for various local monitoring activities.

Ordinal 0 (RD)

This ordinal reserves an EST entry and MST entry to be used by device reconfiguration. No entry needs to be made; the system reserves it automatically.

Ordinal 75_8 (TT)

The system assigns to this equipment a file used for either input from or output to an interactive terminal. This allows the system to determine whether a file requires the special handling needed to accomplish terminal input/output. Bytes 2 and 4, set by the NAMIAF CMRDECK entry, store the number of network terminals and the number of high-speed terminals, respectively.

The TT entry format is:

EQ75=TT,ON,controller,unit,channel_1,channel_2.

Parameter	Description
controller	One octal digit stored in the controller number field of the EST entry.
unit	Number in the range from 0 through 17_8 stored in the unit number field of the EST entry.
channel_i	Number in the range from 0 through 37_8 stored in the corresponding channel field in the EST entry. Bit 53 of the pseudo equipment EST entry is set if pseudo channels are entered (to distinguish channel 0 from no channel); if multiple channels are specified, only the first can be zero.

† The mnemonic appears on the E,A display (refer to the NOS 2 Operator/Analyst Handbook).

Ordinal 76_8 (TE)

If an association is established between file name and volume serial number with an ASSIGN, LABEL, REQUEST, or VSN command, the system automatically enters EQ76 in the file's FNT/FST entry. When a tape having the desired volume serial number is assigned to the file, the system replaces EQ76 in the file's FNT/FST entry with the EST ordinal of the tape unit on which the tape is mounted. If a file which has had the file name and volume serial number association established by a VSN command is returned prior to attempting to assign the tape equipment to the file, the FNT/FST entry is canceled.

The TE entry format is:

EQ76=TE,ON,controller,unit,$channel_1$,$channel_2$,$channel_3$,$channel_4$.

Parameter	Description
controller	One octal digit stored in the controller number field of the EST entry.
unit	Number in the range from 0 through 17_8 stored in the unit number field of the EST entry.
$channel_i$	Number in the range from 0 through 37_8 stored in the corresponding channel field in the EST entry. Bit 53 of the pseudo equipment EST entry is set if pseudo channels are entered (to distinguish channel 0 from no channel); if multiple channels are specified, only the first can be zero.

Ordinal 77_8 (NE)

The system uses 77_8 internally to signify that a file is assigned, but that no space exists on the device. If a read is tried, end-of-information (EOI) status occurs. If a write is attempted, the data is discarded.

The NE entry format is:

EQ77=NE,ON,controller,unit,$channel_1$,$channel_2$,$channel_3$,$channel_4$.

Parameter	Description
controller	One octal digit stored in the controller number field of the EST entry.
unit	Number in the range from 0 through 17_8 stored in the unit number field of the EST entry.
$channel_i$	Number in the range from 0 through 37_8 stored in the corresponding channel field in the EST entry. Bit 53 of the pseudo equipment EST entry is set if pseudo channels are entered (to distinguish channel 0 from no channel); if multiple channels are specified, only the first can be zero.

For example, you can use ordinal 77_8 with the permanent file utility to validate the integrity of a permanent file device, without taking the time to actually create a dump file on tape. In this case, enter the following (refer to the NOS 2 System Maintenance Reference Manual):

```
X.DIS.
ASSIGN,NE,TAPE.
PFDUMP.
```

This causes all dump data to be discarded, even though the permanent file device is read and informative messages about the permanent file device are issued to the system console. These messages are described in the NOS 2 System Maintenance Reference Manual.

SYSTEM DISPLAY CONSOLE EQUIPMENT EST ENTRY

NOS requires that at least one system console is available for use. The format of the entry is:

```
EQeq=DS,status,controller,0,channel.
```

Parameter	Description
eq	EST ordinal of the console; from 1 through 74_8.
DS	System display console equipment.
status	Indicates whether the console is available for system use; enter one of the following values. **status** / **Description** ON — System console is available. OFF — System console can be used only for the DSD commands.
controller	Number of system console controller; from 0 through 7.
channel	Number of the channel to which the console equipment is connected; from 0 through 13_8 and from 20_8 through 33_8.

Example:

The following entry specifies an EST of 10_8 for the system console. The console is available for system use on controller 7 and channel 10_8.

```
EQ10=DS,ON,7,0,10.
```

UNIT RECORD EQUIPMENT EST ENTRY

A unit record equipment EST entry defines card readers, card punches, and line printers.

```
EQeq=type,status,controller,,channel,,id,fc,ps.
                    or
EQeq=type-n,status,controller,,channel,,id,fc,ps.
                    or
EQeq=type-nP,status,controller,,channel,,id,fc,ps.
```

Parameter	Description
eq	EST ordinal of the unit record equipment; from 1 through 74_8.
type	Unit record equipment type; NOS supports the following unit record equipments.

type	Equipment
	Card reader
CR	405-3447/3649
	Card punch
CP	415-3446/3644
CP	415-30
	Line printer
LR	580-12
LS	580-16
LT	580-20

n — Print train for local batch line printer; from 1 through 7. NOS supports the following print trains.

n	Print Train	Description
1	595-1/596-1	CDC graphic 63/64-character set.
5	595-5/596-5	ASCII graphic 63/64-character set.
6	595-6/596-6	ASCII graphic 95-character set.
7	595-6/596-6	ASCII graphic 95-character set used as an ASCII 63/64-character set (as on a 595-5/596-5).

If you set a nonsupported print train value, n defaults to a supported value. If you omit n or specify 2 or 3, the actual value of n is 1. If you specify 4, the actual value is 5.

P — Specifies that a 580 printer is equipped with a programmable format controller. You must append p to the print train entry previously described, and you must specify a print train.

status — Specifies whether unit record equipment is available for system use; enter one of the following values.

status	Description
ON	Unit record equipment is available.
OFF	Unit record equipment is ignored during system operation.

Parameter	Description
controller	Controller number for equipment; from 0 through 7.
channel	Number of channel to which unit record equipment is connected; from 0 through 13_8 and from 20_8 through 33_8.

NOTES

When performing a coldstart, a card reader must be available on channel 12_8 or 13_8.

To ensure that all printers are restored to their original states (such as eight lines per inch and auto page eject) after a master clear has been issued, all unit record equipment should be available on dedicated channels. If it is not, printers revert to six lines per inch, and no auto page eject status after a master clear is issued.

id	One- or two-digit octal numeric identifier assigned to the device; from 0 through 67_8. This id is assigned to any output created by a job. For card readers, all jobs loaded from this card reader are assigned the identifier id.
fc	Two-character optional forms code assigned to a line printer or card punch. If the forms code is not present, the forms code field is cleared. The forms code must either be null (not specified) or in the range from AA through 99.

NOTE

The forms code cannot be assigned to a card reader.

ps	Paper size, L or S, for local batch line printer. Default is L.

L Long (11 inch) paper will be mounted.

S Short (8 1/2 inch) paper will be mounted.

Examples:

```
EQ11=CR,ON,4,,12.
EQ12=CP,ON,5,,12.
EQ20=LR,ON,6,,12,,15,AA.
EQ21=CR,ON,7,,12,,15.
EQ22=LT-6P,ON,2,,12.
```

MAGNETIC TAPE EQUIPMENT EST ENTRY

The released tape subsystem supports a maximum of 16 magnetic tape units. The minimum number of magnetic tape units that NOS requires is two 667s, 669s, 677s, or 679s. The format of the entry is:

EQeq=MT-n,status,controller,unit,$chan_1$,$chan_2$,...,$chan_n$,hwfeature.

or

EQeq=NT-n,status,controller,unit,$chan_1$,$chan_2$,...,$chan_n$,hwfeature.

Parameter	Description
eq	EST ordinal of the tape unit; from 1 through 74_8. Refer to the MT-n or NT-n parameter.
MT-n or NT-n	Equipment type; MT specifies seven-track tape units, and NT specifies nine-track tape units. n is the total number of magnetic tape units connected to the controller, from 1 through 20_8 for 677 and 679 units with a 7021-31/32 controller and from 1 through 10_8 for 667 and 669 units with a 7021-21/22 controller. The system automatically generates n EST entries with consecutive EST ordinals beginning with the ordinal specified in the eq parameter. The n units begin with the unit number specified in the unit parameter.

NOTE

To clear an MT-n or NT-n assignment, enter an EQeq=0 entry for all n units. For example, to clear EQ50=MT-4,ON,..., enter:

```
EQ50=0.
EQ51=0.
EQ52=0.
EQ53=0.
```

Parameter	Description
status	Indicates whether the tape unit is available for access; enter one of the following values:

status	Description
ON	Magnetic tape unit is available for access.
OFF	Magnetic tape unit is ignored during system operation.

Parameter	Description
controller	Controller number for the tape unit; from 0 through 7.
unit	Number of the lowest numbered magnetic tape unit to be processed; units must have consecutive physical unit numbers; from 0 through 7 for 667 and 669 units or from 0 through 17_8 for 677 and 679 units.
$chan_i$	Number of the channel to which the tape unit is connected; from 0 through 13_8 and from 20_8 through 33_8. A controller can be connected to from one through four channels, depending on the controller model. However, a maximum of four channels can be handled regardless of the number of controllers.

Parameter	Description
hwfeature	Hardware features available. This parameter specifies the following hardware characteristics of the tape unit(s) being defined; select one option.

For 677/679 units:

hwfeature	Description
10	The unit(s) being defined cannot process 6250-cpi group-encoded (GE) tapes. If the system detects a unit with this capability, it automatically changes this value to 11, indicating the availability of the GE feature.
11	The unit(s) being defined can process 6250-cpi GE tapes. Specify this value if a unit being defined has the capability of processing GE tapes but is down. In such a case, the system would not be able to connect to the unit to determine the availability of the GE feature.

For 667/669 units:

hwfeature	Description
20	The FCOs needed to implement the block identification feature have been installed in the 7021 controller for the unit(s) being defined. If the controller is a two-channel model, the block identification feature is implemented on both channels.
22	Either the block identification feature is not implemented on the 7021 controller or is implemented on only one channel of a two-channel model. Select this option if the block identification feature is not implemented.

MASS STORAGE FACILITY EQUIPMENT EST ENTRY

The mass storage facility hardware consists of a cartridge storage unit (CSU) and from one through four mass storage transports. The CSU and each transport is represented by a unique EST entry. The EST entries for the transports on a CSU must immediately follow the EST entry for that CSU. There must be no other EST entry between the entry describing a CSU and those describing its transports.

A mainframe that runs MSSEXEC in slave mode must have a CSU equipment type entry, even though no MSF hardware is actually configured for this mainframe.

CSU Entry

The format is:

```
EQeq=CS,status,controller,unit,channel,0,csuid,msamsid,unitmsid.
```

Parameter	Description
eq	EST ordinal of CSU; from 1 through 74_8. Although the software allows an ordinal of 74_8, the ordinal must be lower than that value because the entries for transports must follow the CS entry.
CS	Indicates a CSU.
status	Specifies whether the CSU is available for use; enter one of the following values. **status** — **Description** ON — CSU is available. OFF — CSU is ignored during system operation.
controller	Number of the mass storage coupler (MSC); from 0 through 7.
unit	Number of the mass storage adapter (MSA) port to which the CSU is connected; from 0 through 37_8.
channel	Number of the channel to which the CSU is connected; from 0 through 13_8 and from 20_8 through 33_8.
csuid	Identifier of the CSU; one of the letters A through M.

NOTE

The following two parameters are the MSA and CSU identifiers that are hardwired into the MSA and CSU equipment.

Parameter	Description
msamsid	Number from 0 through 17_8. Obtain this number from the customer engineer. Use 0 if the FCO that supports this parameter is not installed.
unitmsid	Number from 0 through 377_8. Obtain this number from the customer engineer. Use 0 if the FCO that supports this parameter is not installed.

Mass Storage Transport Entry

The format is:

```
EQeq=CT,status,controller,unit,channel,0,position,msamsid,unitmsid.
```

Parameter	Description
eq	EST ordinal of the transport; from 1 through 74_8.
CT	Indicates the mass storage transport.
status	Specifies whether the transport is available for use; enter one of the following values. **status** / **Description** ON — Transport is available. OFF — Transport is ignored during system operation.
controller	Number of MSC; from 0 through 7.
unit	Number of the MSA port to which the transport is connected; from 0 through 7.
channel	Number of the channel to which the transport is connected; from 0 through 13_8 and from 20_8 through 33_8.
position	Physical position of the transport in the CSU; from 0 through 3. Obtain this number from the customer engineer.

NOTE

The following two parameters are the MSA and transport identifiers that are hardwired into the MSA and transport equipment.

msamsid	Number from 0 through 17_8. Obtain this number from the customer engineer. Use 0 if the FCO that supports this parameter is not installed.
unitmsid	Number from 0 through 377_8. Obtain this number from the customer engineer. Use 0 if the FCO that supports this parameter is not installed.

STIMULATOR EQUIPMENT EST ENTRIES

The interactive stimulator is described in the NOS 2 System Maintenance Reference Manual.

The format of the EST entry used by the interactive subsystem during stimulation is:

EQeq=TT,status,controller,1,channel,0,lines.

Parameter	Description
eq	EST ordinal of the interactive stimulator; from 1 through 74_8.
TT	Indicates an interactive stimulator.
status	Specifies whether the interactive stimulator is available for use; enter one of the following values. status / Description ON: Stimulator is available. OFF: Stimulator is ignored during system operation.
controller	Number of the controller; from 0 through 77_8. Refer to description of channel parameter.
channel	Number of the channel; from 0 through 13_8 and 20_8 through 33_8. The channel/controller combination must not have any equipment attached to it.
lines	Number of lines to stimulate; from 1 through 1000_8. If this parameter is omitted, 100_8 is used.

NETWORK PROCESSING UNIT EST ENTRY

The format of the EST entry is:

EQeq=nt,status,controller,pip,channel,0,node,sam.

Parameter	Description
eq	EST ordinal of the NPU; from 1 through 74_8.
nt	Specifies the type of NPU; enter the following value. nt / Description NP: 255x NPU.
status	Specifies whether the NPU is available for use; enter one of the following values. status / Description ON: NPU is available. OFF: NPU is ignored during system operation.

Parameter	Description
controller	Number of the controller for the NPU; from 0 through 7.
pip	Peripheral interface program index, which determines which copy of the PP driver drives this NPU; from 1 through 4. Up through four EST entries can have the same PIP index (that is, one PP can drive four NPUs).
channel	Number of the channel to which the NPU is connected; from 0 through 13_8 and from 20_8 through 33_8.
node	Node number of coupler associated with the NPU being defined; from 1 through 37_8. This value is the same as the NODE parameter on the COUPLER statement in the network configuration file definition. For the procedure to assign this value, refer to the Network Definition Language Reference Manual.
sam	System Autostart Module (SAM) flag. Enter one of the following values. 0 SAM is absent. 1 SAM is present.

NOTE

The node parameter is specified as an octal value on the NPU entry. The NODE parameter is specified on NDL statements as a decimal value.

Example:

Assume that three NPUs exist on channels 4, 5, and 6, all with controller 7. The NPUs are connected to coupler nodes 2, 8, and 11, respectively. The NPU on channel 5 has a system autostart module; the others do not. The first two NPUs are to be driven by the same PP. The EST entries for these NPUs are:

```
EQ70=NP,ON,7,1,4,0,2,0.

EQ71=NP,ON,7,1,5,0,10,1.

EQ72=NP,ON,7,2,6,0,13,0.
```

The NDLP input for the network configuration would include the following statements (refer to the Network Definition Language Reference Manual for a complete description of these statements).

```
CPL1:  COUPLER     NODE=2,HNAME=HOST1.

CPL2:  COUPLER     NODE=8,HNAME=HOST1.

CPL3:  COUPLER     NODE=11,HNAME=HOST1.
```

The node parameter of the EST entry and NODE on the COUPLER statement have the same numeric values, in this case 2, 8 (10_8), and 11 (13_8).

NETWORK ACCESS DEVICE EST ENTRY

The format for the network access device (NAD) EST entry is:

EQeq=NC,status,0,0,channel.

Parameter	Description
eq	EST ordinal of the NAD; from 1 through 74_8.
NC	Indicates a NAD.
status	Specifies whether the NAD is available for use; enter one of the following values:

status	Description
ON	NAD is available.
YES	NAD is ignored during system operation.

Parameter	Description
channel	Number of the channel to which the NAD is connected; from 0 through 13_8 and from 20_8 through 33_8.

TWO-PORT MULTIPLEXER EST ENTRY

The two-port multiplexer (TPM) must be described in the EST to allow use of the remote diagnostic facility (RDF) on models 815, 825, 835, 855, 865, and 875.

The format of the entry is:

EQeq=RM,status,,port,channel.

Parameter	Description
eq	One- or two-digit octal EST ordinal of the TPM; 1 through 74_8.
RM	TPM equipment type.

Parameter	Description
status	Specifies whether the equipment is available for use:

status	Description
ON	Available.
OFF	Equipment is ignored during system operation.

port	Port number to be used by RDF, 0 or 1. RDF normally uses port 1.
channel	Channel number on 815, 825, 835, 855, 865, and 875; channel 15 is required.

MAP III EQUIPMENT EST ENTRY

The MAP III equipment EST entry allows use of the matrix algorithm processor.

The format for the entry is:

```
EQeq=MP,status,0,0,channel.
```

Parameter	Description
eq	EST ordinal of MAP III; from 1 through 74_8.
MP	Indicates MAP III equipment.
status	Specifies whether the MAP III is available for use; enter one of the following values.

status	Description
ON	MAP III is available.
OFF	MAP III is ignored during system operation.

channel	Number of the channel to which the MAP III is connected; from 0 through 13_8 and from 20_8 through 33_8.

EQUIPMENT ASSIGNMENTS: MASS STORAGE

This subsection describes the following EST entries and mass storage equipment assignments.

Equipment	Entry
Mass storage equipment	EQ
Extended memory	EQ

Equipment Assignment	Entry
Peripheral processor buffer area	PPB
Mass storage allocation control	MSAL

Equipment Assignment	Entry
Temporary files device	TEMP
Permanent files device	PF
System library device	SYSTEM
Alternate system library device	ASR
Logical device removal	DOWN
Default family name	FAMILY
Removable device	REMOVE
Initialization entry	INITIALIZE
Link device	LINK
Shared device	SHARE
Independent shared device	ISHARE
Preset link device	PRESET
Preset independent shared device	PRESET
Extended memory allocation	XM
Load buffer controllers	LBC

NOS MASS STORAGE CONCEPTS

Following are descriptions that define the NOS mass storage terminology and summarize the kinds of mass storage equipment assignments that can be specified in the CMRDECK. Table 7-1 summarizes the various functions that a particular mass storage device can serve. For example, if a device is an alternate system device (Device Type column), then it cannot be a system device; it can contain temporary files, direct access files, and indirect access files; it can be a master device or a nonmaster device; it cannot be removable; it can be either an auxiliary device or a family device; and it can be a shared device or a link device.

Alternate System Device

Whereas a system device contains all the routines in the system library, an alternate system device contains copies of selected system library routines. The ASR entry in CMRDECK (refer to ASR - Alternate System Library Device Assignment in this section) specifies which mass storage devices are to be alternate system devices; the *AD LIBDECK entry (refer to LIBDECK, section 10) on the deadstart tape specifies which system library routines are to reside on these alternate system devices. During system processing, the routines on the alternate system device are used instead of the ones on the system device.

This feature allows each routine in the system library to reside on the mass storage device that is most appropriate to the routine's use. For example, instead of using an 844 system device, a routine that is frequently used could use extended memory, which has a faster transfer rate, as an alternate system device.

Table 7-1. Mass Storage Device Functions

Other Possible Functions	Device Type								
	System	Alternate System	Containing Temporary Files	Containing Direct Access Files	Containing Indirect Access Files	Auxiliary	Default Family	Shared	Link
Alternate system device	No	-	Yes	Yes	Yes	Yes	Yes	Yes	Yes
System device	-	No	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Contain temporary files	Yes	Yes	-	Yes	Yes	Yes	Yes	Yes	Yes
Contain direct access files	Yes	Yes	Yes	-	Yes	Yes	Yes	Yes	Yes
Contain indirect access files	Yes	Yes	Yes	Yes	-	Yes	Yes	Yes	Yes
Master device	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Nonmaster device	Yes	Yes	Yes	Yes	No	No	Yes	Yes	Yes
Removable device	No	No	No	Yes	Yes	Yes	No	Yes	No
Nonremovable device	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Auxiliary device	Yes	Yes	Yes	Yes	Yes	-	No	Yes	Yes
Family device	Yes	Yes	Yes	Yes	Yes	No	-	Yes	Yes
Shared device	Yes	Yes	Yes	Yes	Yes	Yes	Yes	-	-
Link device	Yes	Yes	Yes	Yes	Yes	Yes	Yes	-	-

Alternate Permanent File Family

More than one permanent file family can exist in a system: one default permanent file family and one or more alternate permanent file familes. One permanent file family is defined as the default family by the CMRDECK FAMILY entry (refer to FAMILY - Default Family Name Assignment in this section). If another system's permanent file family is introduced, it is an alternate permanent file family; it can be added without interrupting the default permanent file family's operation.

This is a useful feature if a site has more than one system or has groups of installations. If one system fails, its permanent files can be accessed from another system.

As an example, a site with two systems might run with the mass storage configuration shown in table 7-2.

Table 7-2. Mass Storage Configuration for Two Systems at One Site

System	Ordinal	Device	Spindles	Access Used	Contents
X	1	844	2	A	Direct access files
Y	1	844	2	B	Direct access files

If system Y became inoperative, the B access could be connected to system X. This could be done without interrupting operations of system X.

The CMRDECK entries in system X would be:

Entries	Comments
EQ1=DI-2,ON,0,0,0,3.	Defines access A.
EQ2=DI-2,OFF,0,0,1,10.	Defines access B.
REMOVE=2.	Allows introduction of access B into system X during operation.

The CMRDECK entries in system Y would be:

Entries	Comments
EQ1=DI-2,ON,0,0,1,10.	Defines access B.
EQ2=DI-2,OFF,0,0,0,3.	Defines access A.
REMOVE=2.	Allows introduction of access A into system Y during operation.

To allow for introduction of an alternate permanent file family:

1. Define the equipment to be introduced or removed in the CMRDECKs for both systems (refer to the previous example).

2. Specify all of the equipment that may be introduced or removed during system processing as removable.

3. When you want to introduce the equipment into a system, use the ON operator command to indicate that the equipment that was set to the OFF position in the system in operation is now available. This introduces the alternate permanent file family.

4. Enable the validation files for the family by entering X.ISF(FM=familyname). Refer to the NOS 2 Operator/Analyst Handbook.

Auxiliary Device

An auxiliary device is a mass storage device that is not part of a family. It is a supplementary permanent file storage device, which may be privately owned (PRIVATE) or may be shared by many users (PUBLIC). An auxiliary device resides on either a removable or nonremovable device. On the permanent file entry (PF) for an auxiliary device (for both a removable device and a fixed device without packs), a pack name is specified instead of a family name.

Refer to the NOS 2 Reference Set, Volume 3, for a detailed description of private and public auxiliary devices.

As an example, four 844 spindles to be used as a public auxiliary device could be defined as follows:

```
EQ3=DI-4,ON,0,1,2.
PF=3,X,name.
```

Private auxiliary devices can be created only after the system is up and running. An operator can make a public device a private device by entering the INITIALIZE command with the UN and TY=X parameters (refer to the NOS 2 Operator/Analyst Handbook).

Family Device

A family device is a mass storage device that is part of a family. It can be either a removable device or a nonremovable device. The only difference between the two is that a nonremovable device containing permanent files can also contain a copy of the system library and/or temporary files. Refer to Alternate Permanent File in this section.

On the PF entry, the family name is only important if two systems' permanent files are to run on the same system. A user can only use one family of permanent files; if the user does not specify one, the default FAMILY entry is used.

A family device can contain direct and/or indirect access files. These files are defined in the NOS 2 Reference Set, Volume 3. The files that are allowed are set by the device mask and secondary mask on the PF entry.

Link Device

Either extended core storage or extended semiconductor memory is the medium through which several computer systems are linked to form a multimainframe operating environment (shared MMF). The link device contains the information necessary for the orderly management of the mass storage that can be shared by more than one mainframe. For a description of shared mass storage, refer to SHARE: Shared Device Designation and ISHARE: Independent Shared Device Designation in this section.

Examples:

An ECS to be used as a link device could be defined as follows:

```
EQ11=DE,ON,1000.
LINK=11.
```

Use of DDP could be defined as follows:

```
EQ11=DP,ON,1000,27.
LINK=11.
```

Master Device

The master device contains all of the permanent file catalog entries and indirect access files for a specific user. If permanent file access is required, the user's master device must be available on the system, unless all access is to be to an auxiliary device. The user index (refer to the NOS 2 System Maintenance Reference Manual) and family name uniquely describe a user's master device.

Each master device is organized into five logical sections.

- Allocation information.

  A master device, like all mass storage devices, maintains device labels and track reservation tables (TRTs).

  The device label contains information describing the device, such as family name and user mask, as well as locations of permit and catalog information and indirect access files. Refer to the INITIALIZE entry.

  The TRT is the key to allocating information on the master device and to describing the physical layout of data on the device. Refer to APRDECK, section 8, and to the NOS 2 System Maintenance Reference Manual.

- Catalog information.

  Catalog entries are used to determine the locations and attributes of permanent files. The catalogs for a master device are allocated to contain catalog entries for a specific group of user indexes. A particular catalog track may contain entries for many users, the number depending upon the number of catalog tracks defined for the device. The user index provides the mechanism for differentiating between user's files on a particular catalog track. Refer to the NOS 2 System Maintenance Reference Manual for a more detailed description.

- Permit information.

  A user can explicitly or implicitly allow other users to access her/his permanent files. Refer to the PERMIT command in the NOS 2 Reference Set, Volume 3. Information describing the permission for all permanent files is in the permit file. Catalog entries contain a relative sector address within this permit file for permissions that have been granted for the file.

- Indirect access files.

  The master device contains all of the user's indirect access files. These files can be accessed by commands that generate working copies for manipulation by the user.

- Direct access files.

  Direct access files can reside either on the master device or on another device in the family, depending on the device masks specified on the PF entries. Direct access files are files that can be accessed at their location on mass storage. A working copy is not generated, so any updates or alterations made to the files are permanent.

Multispindle Device

To accommodate files that are larger than one device, you can specify multispindle device assignments. Up to eight spindles of 844 disk drives or up to three spindles of 885 disk drives can be included in the equipment definition of one logical device, when the device is first defined. All spindles must be available for access whenever the device is accessed.

Multispindle devices are treated as one logical device, having a track size equal to n times the single-spindle track size (n is the number of spindles in the device). The tracks of an n-spindle device are broken down into n equally sized segments, each having a length equal to the single-spindle track size. Each segment is contained on a different physical unit.

844 Expander

A nonexpanded controller can have up to eight disk drives connected to it. Each of the connection paths is called a port and is identified by a port number ranging from 0 through 7. An expander (10304 extender) is a hardware device that can be connected between controllers and 844 disk drives to increase the number of disk drives that each controller can access.

The expander can be used only with 844-21 drives, although all equipment definitions and equipment driving software support the 64-drive addressing scheme for both 844-21 (DI/DK) and 844-41 (DJ/DL) type equipment.

Each expander consists of either two or four expansion elements. An expansion element connects to a single controller port and forms a connection path from that port to from one through eight disk drives. The connection paths between an expansion element and the eight possible disk drives are called ranks and are identified by a rank number ranging from 0 through 7. Two expanders with four expansion elements each can be connected to a single controller to allow that controller to access a maximum of 64 disk drives. Each expansion element, however, is logically independent and, as such, could be connected to any port of any controller.

A single controller maximum configuration can be visualized as an eight- by eight-square checkerboard with each square representing one of 64 disk drives (figure 7-1).

Controller Port

Expander Rank	0	1	2	3	4	5	6	7
0	00	01	02	03	04	05	06	07
1	10	11	12	13	14	15	16	17
2	20	21	22	23	24	25	26	27
3	30	31	32	33	34	35	36	37
4	40	41	42	43	44	45	46	47
5	50	51	52	53	54	55	56	57
6	60	61	62	63	64	65	66	67
7	70	71	72	73	74	75	76	77

Unit Numbers (octal)

Port Digit

Rank Digit

Figure 7-1. Expander Addressing Map

A column of squares in figure 7-1 represents all the drives that are accessed through a single controller port. A row of squares represents all the drives that have the same expander rank. Each disk drive that can be accessed by the controller is addressed by a 6-bit unit number. The rightmost 3 bits of this unit number select to which of the eight controller ports the drive is connected. The leftmost 3 bits of the unit number select to which of the eight ranks on an expansion element the drive is connected. This unit number is specified as a two-digit octal number in the mass storage equipment EST entry (refer to EQ - Mass Storage Equipment EST Entry). The right digit (port digit) of the unit number is the port number and the left digit (rank digit) is the rank of the unit in the particular expansion element.

If two disk drives are vertically adjacent on the expander addressing map (figure 7-1), their unit numbers are considered to be vertically ordered; that is, both drives are connected to the same expansion element, both have the same port number, and their rank numbers differ by one (refer to example 1).

If two disk drives are horizontally adjacent on the expander addressing map (figure 7-1), their unit numbers are considered to be horizontally ordered; that is, both drives have the same rank number, and their port numbers differ by one (refer to example 2). The special case of rank numbers of 0 for two horizontally adjacent drives is equivalent to the definition of consecutive unit numbers for other equipment.

All drives connected to a controller, either directly or through an expansion element, are supported as single-unit or multiunit logical devices. Unit numbers can range from 0 through 77_8, rather than from 0 through 7, as for other equipment. Thus, a maximum of sixty-four 844 disk drives connected to a single controller can be addressed. However, a maximum of eight units can be specified per multiunit device. In addition, all units of a multiunit device must be connected to the same channel and, therefore, to the same controller.

Figure 7-2 illustrates a configuration in which two expansion elements and 20 disk drives are connected through one controller. An expansion element with eight drives is connected to port 0, an expansion element with six drives is connected to port 1, and six drives are connected to six ports (ports 2 through 7). Each disk drive is shown as a square with its appropriate unit number inside. This configuration is used in the following three examples to illustrate multiunit device assignments. The controller is assumed to be connected to channel 1. Refer to EQ - Mass Storage Equipment EST Entry for specific information on assigning these devices.

		Controller Port							
		0	1	2	3	4	5	6	7
Expander Rank	0	00	01	02	03	04	05	06	07
	1	10	11						
	2	20	21						
	3	30	31						
	4	40	41						
	5	50	51						
	6	60							
	7	70							

Figure 7-2. 844 Expander Configuration with 20 Drives

NOTE

The following examples illustrate multiunit device assignment of devices with horizontally or vertically adjacent units. Ordered unit numbering is not required if the second form of the mass storage EST entry is used as the CMRDECK entry (refer to EQ - Mass Storage Equipment EST Entry).

Example 1:

Figure 7-3 illustrates a possible configuration for a three-unit vertically adjacent multiunit device. This device could be assigned in the CMRDECK, specifying equipment 2, as:

```
EQ3=DI-2,ON,0,40,1.
        ↑ ↑  ↑
        │ │  └──── Channel
        │ └─────── First Unit
        └───────── Vertically Adjacent Flag
```

The lowest physical unit number of the vertically adjacent device is specified in the unit parameter.

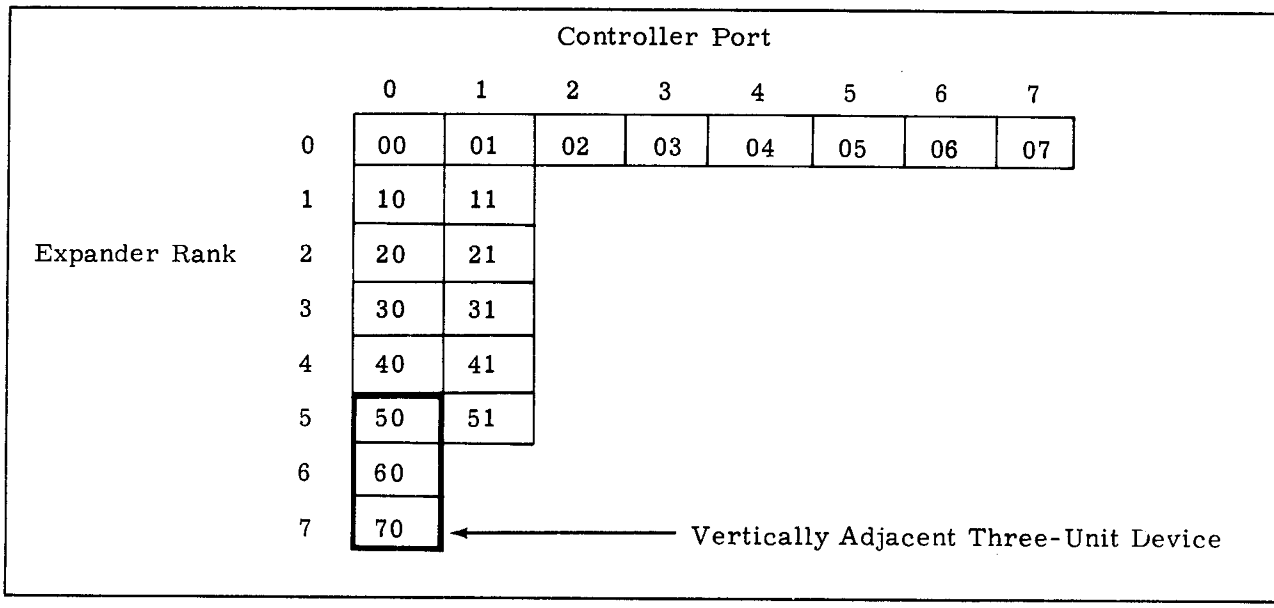

Figure 7-3. Vertically Adjacent Three-Unit Device

Example 2:

Figure 7-4 illustrates a possible configuration for a two-unit horizontally adjacent multiunit device. This device could be assigned in the CMRDECK, specifying equipment 3, as:

```
EQ2=DI-3,ON,4,50,1.
```

Channel
First Unit
Horizontally Adjacent Flag

The lowest physical unit number of the horizontally adjacent device is specified in the unit parameter.

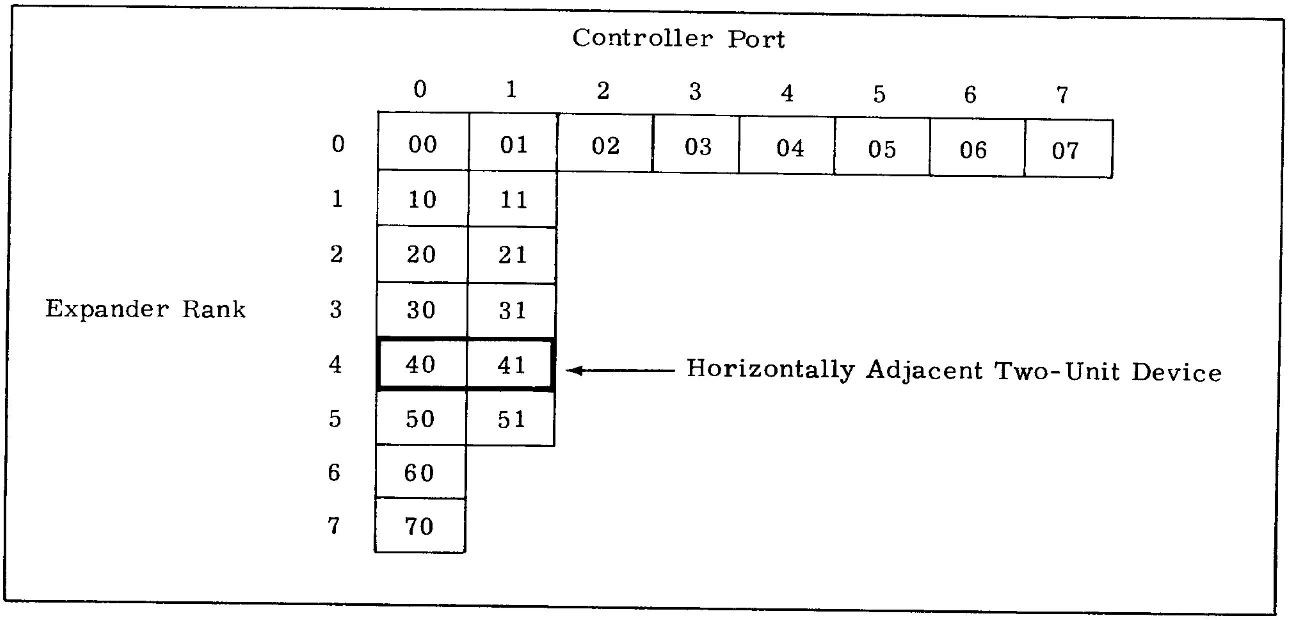

Figure 7-4. Horizontally Adjacent Two-Unit Device

Example 3:

Figure 7-5 illustrates a possible configuration of 20 disk drives into seven devices. These devices could be assigned in the CMRDECK as:

```
EQ1=DI-1,ON,4,60,1. (vertically adjacent)
EQ2=DI-1,ON,4,70,1. (vertically adjacent)
EQ3=DI-2,ON,0,50,1. (horizontally adjacent)
EQ4=DI-4,ON,4,10,1. (vertically adjacent)
EQ5=DI-4,ON,4,11,1. (vertically adjacent)
EQ6=DI-4,ON,0,0,1.  (horizontally adjacent)
EQ7=DI-4,ON,0,4,1.  (horizontally adjacent)
```

Equipments 1 and 2 are defined as vertically adjacent units. This allows them to be on-line initialized into a vertically adjacent two-unit device if they are also defined as removable.

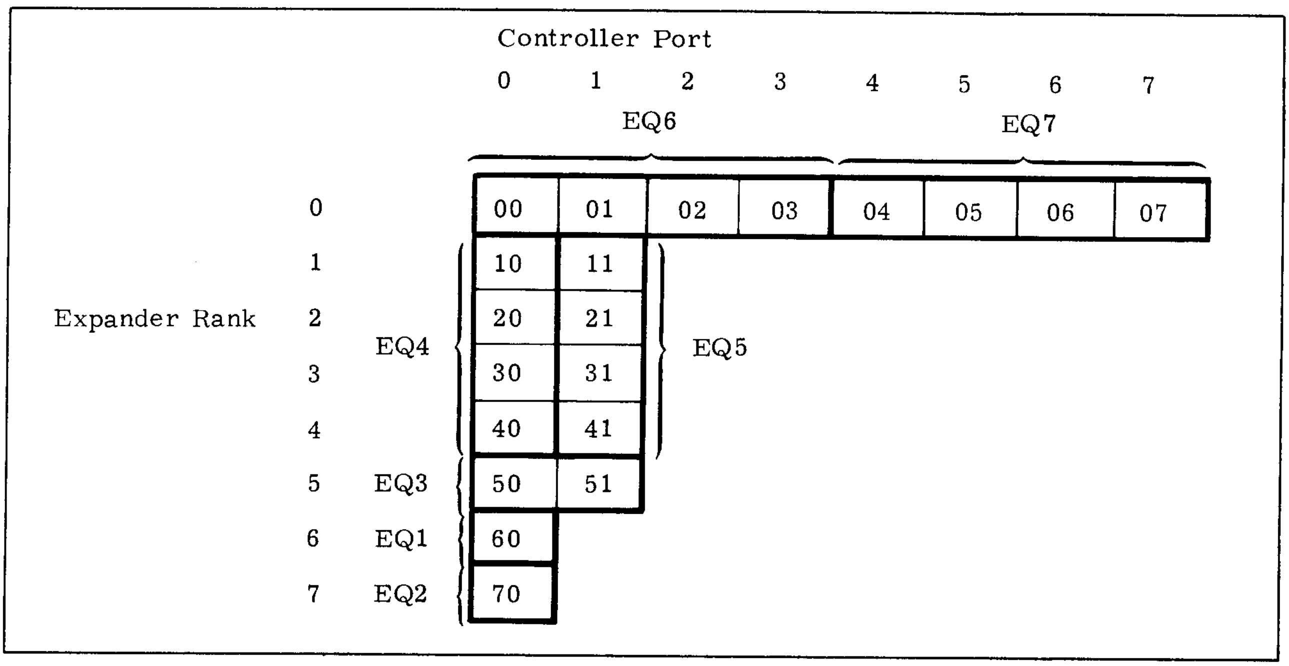

Figure 7-5. Hardware Configured into Seven Devices

Nonremovable Device

A nonremovable device cannot be physically removed during system operation. It can contain a copy of the system library, which means it is a system device; it can also be available for temporary files. It may or may not contain permanent files.

Removable Device

A removable device can be logically or physically added or removed during system operation without causing system malfunction.

A device is specified as removable with the REMOVE entry in the CMRDECK. During deadstart, a removable device is recovered just as is any other mass storage device, if the status is on. If the device is not available, then the status is displayed for the operator (E,M display).

Removable devices can contain permanent files but cannot contain the system library or temporary files, because a device containing active files (such as temporary or library files) cannot be removed from the system. It can be either an auxiliary device or an alternate permanent file family device.

Shared Device

A shared device contains permanent files that can be accessed by more than one mainframe. To have these permanent files accessible to the mainframe, the device must be defined as shared in the mainframe. Refer to SHARE: Shared Device Designation and ISHARE: Independent Shared Device Designation in this section.

System Device

The system device is a nonremovable device on which the system library resides. It can also contain permanent and temporary files.

Temporary File Device

The temporary file device is a nonremovable device on which temporary system files reside. They include:

Library files.
Local files.
Queued files.
Rollout files.
System files.
Timed/event rollout files.

Buffered Mass Storage Devices

Buffered mass storage devices are those devices for which the system provides an extended memory data buffer. For the model 176 only, an 819 mass storage device with the buffer in LCME is a buffered device. For all models except model 176, 815, 825, 835, and 855, the buffered device is an 885-42 mass storage subsystem with the buffer in ESM. Using extended memory as a data buffer provides the following capabilities.

- The system treats the buffer as a disk cache so that multiple requests for a particular disk data block can potentially be satisfied by doing only one disk read.

- The system buffers data to maintain maximum transfer rates regardless of the user's buffer size.

To maintain maximum transfer rates, a CPU program that resides in CPUMTR controls the buffered device I/O request processing. The system maintains the standard disk I/O interface to the user's programs.

The buffered device error reporting process logs read/write errors in the binary maintenance log but not in the system error log or dayfile. When an unrecovered read/write error occurs, the error message flashes at the system control point. The binary maintenance log must be processed by the Hardware Performance Analyzer (HPA) to get detailed information concerning buffered device errors.

The following considerations are important to system performance.

- The amount of I/O buffer space in extended memory. Refer to XM - Extended Memory Allocation.
- The size of the hash buffer parameter. Refer to EQ - Mass Storage Equipment EST Entry.
- The amount of space reserved in central memory for providing PP access to buffered devices. Refer to PPB - Peripheral Processor Buffer Area Entry.

EQ — MASS STORAGE EQUIPMENT EST ENTRY

The purpose of the mass storage equipment (EQ) entries is to describe all mass storage peripheral equipment. NOS requires that at least 6 million words of mass storage be available.

There can be up to 31 logical mass storage devices, and therefore, up to 31 mass storage EST entries (this number does not include ordinal 0, which is automatically defined by the system and is used by the on-line mass storage reconfiguration routines). An entry, however, can refer to more than one physical unit. For example, two 844 spindles can be defined as either two logical devices with two EQ entries or as one logical device with one EQ entry.

A unit is a dual-access unit if it is accessed by one mainframe through two different controller-channel access routes. To define a unit as a dual-access unit with its EQ entry, specify two channel parameters. The channels should be from two controllers. Only one channel of a dual-channel access controller is recommended for use on a single mainframe, since using both channels of the controller results in a performance degradation rather than an improvement. Therefore, if both channel accesses of a controller are physically connected to the same mainframe, you should define only one of them on an EQ entry.

NOTE

A device's EQ entry must precede any of the following entries for that device: ASR, TEMP, MSAL, REMOVE, PF, SYSTEM, FAMILY, INITIALIZE, SHARE, ISHARE, DOWN (refer to the NOS 2 Operator/Analyst Handbook), XM, and LINK. If you redefine a device's EQ entry, then you must also redefine those entries.

There are three forms of the entry. One form is for extended memory (refer to EQ - Extended Memory EST Entry in this section). The other two forms of the entry are:

EQeq=type-n,status,controller,unit,$chan_1$,$chan_2$,apr,hb.

or

EQeq=type-Nn,status,0,u_1,...,u_n,$chan_1$,$chan_2$,apr,hb.

Parameter	Description
eq	EST ordinal of the mass storage equipment; from 1 through 37_8. This range depends upon the value of NMSD, the number of mass storage devices, which is set when the system is assembled (refer to PPCOM Parameters in section 6).
type-n or type-Nn	Equipment type with n units. n is the number of units connected to a controller. The following mass storage equipment is supported by NOS.

type	Equipment	Number of Units
DI	844-21, 7054/7154 (half-track).	1 through 8
DJ	844-41/44, 7054/7154 (half-track), 7155-1.	1 through 8
DK	844-21, 7154 (full-track).	1 through 8
DL	844-41/44, 7154 (full-track), 7155-1.	1 through 8
DM	885-11/12, 7155-1 (half-track).	1 through 3
DQ	885-11/12, 7155-1 (full-track).	1 through 3
DB†	885-42, 7155-401(full track)	1 through 3
DV	Single density 819	1
DW	Double density 819	1

You can define the 844 and 885 physical units with a separate EQ entry for each, or if more continuous storage is needed than is possible with one unit, you can define more than one physical unit as one logical device with one EQ entry. If you use the first form of the EQ entry, n is the number of adjacent numbered units defined by the EQ entry, and unit identifies the lowest numbered unit of the n units. If you use the second form of the EQ entry, n is the number of units composing the device as specified by u_1 through u_n.

For example, two 844 units (0 and 1) to be accessed as two units are defined as:

```
EQeq=DI-1,status,0,0,chan1,,apr.
EQeq=DI-1,status,0,1,chan1,,apr.
```

†Not applicable for models 815, 825, 835, and 855.

Parameter	Description

	Two 844 units (0 and 1) to be accessed as one logical unit are defined as:
	
	`EQeq=DI-2,status,0,0,chan1,,apr.`
	or
	`EQeq=DI-N2,status,0,0,1,chan1,,apr.`
	
	An advantage to accessing the two units as one logical unit is that less space is used in CMR (650_8 words for the 844-21). A disadvantage is that if either unit malfunctions or is destroyed, both units are affected.

status	Specifies whether or not the equipment is available for access; one of the following values. **status** / **Description** ON — Equipment is available. OFF — Equipment is not accessed during system operation. Specify OFF if the equipment is malfunctioning and access is not desirable. If the equipment is removable and is not available at deadstart, the system determines that it is unavailable, even if its EQ status entry specifies ON. If INITIALIZE is entered, the equipment is not initialized until it is set to ON status. During system operation, the operator can initiate access to this device by entering the ON command. If the equipment can be used with either one of two different systems (removable devices, not dual access), define the status of the the EQ entry as ON in the system to which it is currently available for access; define the status of the EQ entry as OFF in the system to which it is not currently available for access.
controller	Orientation flag for the 7054/7154; one of the following values. **Value** / **Description** 0 — Units are horizontally adjacent. 4 — Units are vertically adjacent. Refer to figures 7-3, 7-4, and 7-5 for examples of this parameter for 844 equipment.
unit	Unit number, from 0 through 77_8 for all except 819 units. 819 units work only with model 176; unit number is either from 0 through 3, if on FLPP channels 2 and 3, or from 4 through 7, if on FLPP channels 6 and 7. If the EQ entry is defining more than one unit of a multispindle device, unit refers to the lowest numbered unit of the n units that have adjacent physical unit numbers. Refer to the type-n parameter description.

Parameter	Description
$u_1,\ldots,u_n$	One- or two-digit unit number(s) composing the logical device. The number of unit numbers specified must equal the number of units in the device as specified by the type-Nn parameter previously described. You can specify from one through eight units.

For example, three 844 units (41, 20, and 6) accessed as a single logical device are specified with the following entry.

```
EQeq=DI-N3,status,0,41,20,6,chan1,,apr.
```

Parameter	Description
$chan_1$	For all devices except 819, number of the channel or channels to which the controller is connected; from 0 through 13_8 and from 20_8 through 33_8. For 819s, the channel pair for input/output (I/O) multiplexer (primary access); one of the following values.

$chan_1$	Description
2	Channels 2 and 3.
4	Channels 4 and 5.
6	Channels 6 and 7.

Parameter	Description
$chan_2$	For all except 819s, indicates dual-access unit; cannot be 0; $chan_1$ and $chan_2$ should be connected to different controllers. $Chan_2$ is assigned first if it is free. For 819s, the secondary access channel pair; one of the values described for $chan_1$.
apr	One- through four-digit octal number that indicates which APRDECK to use. If apr is omitted, the first APRDECK, APR0000, is assumed.
hb	Length of hash table segment for this unit; must be a power of 2 and from 10_8 through 400_8. The hash table is maintained in extended memory. The hash table keeps track of the extended memory copies of disk segments for DV, DW, and DB disk units. The hb parameter should be as large as possible for maximum efficiency. If the hash buffer is too small, the system's ability to recognize disk data already in buffers is diminished, thus causing unnecessary disk reads.

EQ — EXTENDED MEMORY EST ENTRY

Three possible extended memory configurations exist. The first configuration is for ECS, LCME, and ESM. It allows a CPU (or two, if dual CPUs are available) to communicate with extended memory. The second configuration is the same as the first except additionally it uses the distributive data path (DDP) or low-speed port (LSP). This configuration allows PPs in the system, as well as the CPU, to communicate directly with ECS or ESM. It does not apply to LCME. NOS supports up to two DDPs or LSPs per mainframe. (Refer to User ECS in section 9 and Extended Memory for User Access Entry in this section for information regarding user-access extended memory.) For the third configuration, part of physical memory is used as extended memory (UEM) on models 815, 825, 835, 855, 865, and 875.

NOTE

If extended memory is not included in the hardware configuration, do not make an extended memory EST entry.

The format of the entry is:

EQeq=type,status,MA,exmem,ddp,size,$chan_1$,$chan_2$,apr,$chan_3$.

Parameter	Description
eq	EST ordinal of extended memory; from 1 through 37_8. This range depends on the value of NMSD (refer to PPCOM Parameters in section 6).
type	Extended memory equipment type; one of the following values. type — Description DE — DDP or LSP is not available. DP — DDP or LSP is available.
status	Specifies whether extended memory is available for access; one of the following values. status — Description ON — Extended memory is available. OFF — Extended memory is ignored during system operation.
MA	Maintenance mode. If you specify MA, online extended memory diagnostics are allowed to reference the half of extended memory that is placed in maintenance mode at the controller. The other half of extended memory is available to the system. The size of available physical extended memory is divided by 2 at deadstart. When you initially place an extended memory device in maintenance mode, all mainframes using the extended memory must initialize it. When you place ECS in maintenance mode, you must also make the PRESET entry for multimainframe operation. Refer to INITIALIZE and PRESET in this section and in the NOS 2 Operator/Analyst Handbook. If you do not specify MA, do not enter its trailing comma; if the comma is present, an error message is issued.

Parameter	Description
exmem	Type of large memory; one of the following values. If you do not specify exmem, E1 is assumed.

exmem	Description
E1	ECS I for all mainframes except models 176, 815, 825, 835, and 855.
E2	ECS II or ESM for all mainframes except models 176, 815, 825, 835, and 855.
LE	LCME only for model 176.

exmem	Description
EM	UEM for models 815, 825, 835, 855, 865, and 875. Type must be DE. The system ensures that the sum of memory words specified by the MINCM CMRDECK entry and specified by this entry for UEM is present.
ES	ESM accessed through maintenance port.

If you do not specify exmem, do not enter its trailing comma.

ddp — Type of DDP; one of the following values. If you do not specify ddp, D1 is assumed if type is DP (if type is DE no value is assumed).

ddp	Description
D1	DC135 DDP.
D2	DC145 (parity enhanced) DDP.

If you do not specify ddp, do not enter its trailing comma.

size — Size of extended memory; one of the values in the following table for ECS I. For ECS II and LCME, size ranges from a minimum value of 10_8, which specifies 4K (K represents 1024 words) through a maximum value of 10000_8, which specifies 2048K. For ESM, the minimum value is the same as for ECS II. However, the maximum value is 100000_8, which specifies 16384K. The system uses space beyond 2048K for buffers for 885-42 devices connected to 7155-401 controllers only. For UEM, size ranges from a minimum value of 10_8, which assigns 4K for UEM, through a maximum value of 10000_8-MINCM, which assigns 2000K-MINCM (convert MINCM to decimal) for UEM.

(Octal) Size	ECS I Available	Number of Banks
400	125K	1
1000	250K	2
2000	500K	4
4000	1000K	8
10000	2000K	16

Parameter	Description
$chan_1$,$chan_2$	Numbers of the channels to which the DDP is connected; from 0 through 13_8 and 20_8 through 33_8. If the equipment type is DE, do not specify a channel parameter. If you do, the system recognizes the DE entry as a DP entry. If the equipment type is DP, specify either one or two channels. The second channel cannot be 0. If a DDP is present, the loading of CPU programs residing in ECS or ESM still occurs via the CPU. A DDP must be connected to a channel by itself.
apr	One- to four-digit octal number that indicates which APRDECK to use. If you omit apr, the first APRDECK, APR0000, is assumed.
$chan_3$	Number of the channel to which the maintenance port is connected; from 0 through 13, and 20_8 through 33_8. This channel is for ESM only.

Examples:

```
EQ4=DE,OFF,1000,21.
EQ5=DP,ON,2000,2,3,8.
EQ11=DE,ON,MA,E2,1000.
EQ11=DP,OFF,E1,D2,2000,2,,5.
EQ11=DP,ON,ES,D2,10000,5,6,,7.
```

PPB — PERIPHERAL PROCESSOR BUFFER AREA ENTRY

The PPB entry reserves space in central memory for accessing files on 819 disks or extended memory.

The format is:

PPB=number.

Parameter	Description
number	The number of words/102_8 reserved for PP I/O. It must be sufficient to handle the demand for PP access to the 819 device I/O handler.

MSAL — MASS STORAGE ALLOCATION CONTROL ENTRY

The MSAL entry assigns job files of the specified type to the mass storage devices defined by the specified EST ordinal.

The format is:

MSAL,t=eq_1,eq_2,...,eq_n

Parameter	Description
t	File type; one of the following values.

t	Description
B	LGO files.
D	User dayfiles.
I	Input files.†
L	Local files.
O	Output files.†
P	Primary files.
R	Rollout files.
S	Secondary rollout files.
T	Temporary files (as described for the TEMP entry).

Parameter	Description
eq_i	EST ordinal of a nonremovable mass storage device.

Secondary rollout files are rollout files whose size in sectors is smaller than a threshold specified by the IPRDECK or DSD entry SRST=n (refer to section 9). These files are rolled out to devices specified by the MSAL,S= entry. All files selected for rollout that are equal to or greater in size than the threshold are rolled out to devices specified by the MSAL,R= entry. The default value of SRST is 0; thus, no secondary rollout files exist unless this value is changed. A possible use of this feature is:

Entries	Comments
CMRDECK	
.	
.	
EQ11=DP,ON,1000,27.	Specify equipment 11 as ECS with a DDP.
.	
.	
MSAL,S=11.	Direct secondary rollout files to ECS.
IPRDECK	
.	
.	
SRST=20.	Set threshold count so that large rollout files are rolled out to ECS.

†Routing of a file to queues changes a file's type, not its residency. Thus specifying an MSAL,I=cq or MSAL,O=eq entry does not necessarily force all input or output queue files to the specified devices.

TEMP — TEMPORARY FILES DEVICE ASSIGNMENT ENTRY

To assign a nonremovable mass storage device as available for temporary files, add a TEMP entry for that device to the CMRDECK. Do not add a TEMP entry for a device with an EQ entry that already has a REMOVE assignment. Temporary files include:

Library files	Rollout files
Local files	System files
Queued files	Timed/event rollout files

The format is:

TEMP=eq_1,eq_2,...,eq_n.

Parameter	Description
eq_i	EST ordinal of nonremovable mass storage device; from 1 through 37_8. This range depends upon the value of NMSD (refer to PPCOM Parameters in section 6). One or more ordinals can be specified with one entry.

PF — PERMANENT FILES DEVICE ASSIGNMENT ENTRY

Before initializing a mass storage device (with the INITIALIZE entry in CMRDECK), add a PF entry for that device to the CMRDECK anywhere after the device's EQ entry. The PF entry information becomes part of the device's label when it is initialized; this label is recovered during subsequent deadstarts. For subsequent deadstarts, it is not necessary that the PF entry be part of the CMRDECK on the deadstart tape; if it is, it is ignored.

If the unit is a family device, the format is:

PF=eq,type,dm,sm,name,device,nc.

If the unit is an auxiliary device, the format is:

PF=eq,type,name,nc.

Parameter	Description
eq	EST ordinal of the device; from 1 through 37_8. This range depends upon the value of NMSD (refer to PPCOM Parameters in section 6).
type	Type of device; one of the following values.

type	Description
F	Family device. It can contain indirect access files if the dm parameter is within the range 1 through 377_8. It can contain direct access files if the sm parameter is within the range from 1 through 377_8. It is a master device if the dm parameter is specified.
X	Auxiliary device, which can contain both direct and indirect access files. X must be specified on a unit's PF entry if any of the auxiliary device commands are to be used for the device.

Refer to table 7-3 for dependencies.

Table 7-3. PF Entry

Type of PF Device	Files Permitted on Device	PF Entry Parameter Settings				
		type	dm	name	device	sm
Auxiliary †	Indirect and/or direct	X	Omit	pack	Omit	Omit
Family	Direct only	F	0	family	1 - 77_8	1 - 377_8
	Indirect only (master device)	F	1 - 377_8	family	1 - 77_8	0
	Indirect and direct (master device)	F	1 - 377_8	family	1 - 77_8	1 - 377_8

†If a user name is specified for an auxiliary device, enter the INITIALIZE command after deadstart. Refer to the NOS 2 Operator/Analyst Handbook for the procedure.

Parameter	Description
dm	Specifies the unit's device mask; from 0 through 377_8. Set according to information in the NOS 2 System Maintenance Reference Manual. Omit this parameter if the device is an auxiliary device. The device mask for a permanent file device defines the groups of users whose catalogs reside on the device for a particular family.
sm	Specifies the unit's secondary mask; from 0 through 377_8. Set according to information in the NOS 2 System Maintenance Reference Manual. Omit this parameter if the device is an auxiliary device. This parameter controls the residence of direct access files in the same way that dm controls the residence of indirect access files.
name	Designates either the name of the family to which the unit belongs or its pack name if it is an auxiliary device; from one through seven alphanumeric characters. Do not use the family name 0: it is reserved. The family name describes the permanent file devices available to a user.† A family may consist of from 1 through 63 logical devices; however, the master devices within the family must have device masks totaling 377_8 if all possible user indexes are to be accommodated.

†If not otherwise specified, the default family name becomes part of the tape label information. It is checked and verified if the user specifies the FA=A parameter on a command. Refer to the NOS 2 Reference Set, Volume 3 for a discussion of FA=A.

Parameter	Description
	Usually a system runs with one family of permanent file devices available. But you can activate additional families on a system, in order to allow the users of these families to access their permanent files through an alternate system. This might be helpful if one system supplies backup service to another system. When more than one family is active on a system, users with matching user indexes access the same permanent files on a public auxiliary device. You can avoid this situation by predetermining a range of user indexes for each family running on a system. When a new family is introduced into a system, its user indexes should be checked against those of the family or families currently running and any matching indexes should be changed.
	The pack name is the unique seven-character name associated with an auxiliary device. An auxiliary device is a self-contained permanent file device: all permanent files (whether direct or indirect access) represented by the catalogs on the device reside on that device. To access a file on an auxiliary device, users must specify the pack name as part of the permanent file request. The pack name is used instead of the usual algorithm for determining catalog location (user masks and family name). An auxiliary device can be private or public. Any user who knows the pack name and has the appropriate permissions and validations can access files on an auxiliary device.
device	Number of the device in the family; from 1 through 77_8. Omit this parameter if the device is an auxiliary device.
	A permanent file that does not reside on the master device has a device number in the catalog entry or on the master device. The device number specifies on which alternate device within the family the file resides.
nc	Number of catalog tracks (optional) used only for master devices; from 1 through 200_8. This value must be a power of 2. If you do not specify nc, one of the following default values (based on the equipment type) is supplied.

Default nc	Equipment	Type
40	844-21	DI/DK
40	844-41/44	DJ/DL
10	885-11/12	DM/DQ
10	885-42	DB
10	819	DV/DW
4	Extended memory	DE
4	ECS or ESM with DDP	DP
1	Private device	

Examples:

```
PF=2,F,125,125,SYSTEM,3,200.
PF=17,X,PACK.
```

SYSTEM — SYSTEM LIBRARY DEVICE ASSIGNMENT ENTRY

The SYSTEM entry specifies which mass storage devices are to contain copies of the NOS system library from the deadstart tape. A system device can be any mass storage device as well as extended memory.

Throughput can be greatly improved by specifying more than one system device. For example, if two system devices are specified and they are on different channels, the time required to access system programs can be reduced. When the channel for one system device is busy, the other is accessed.

The following restrictions apply.

- The EQ entry for a system device cannot have the status set to OFF.
- A REMOVE entry cannot exist in the CMRDECK for a device being specified as a SYSTEM device.
- If more than one device is specified as a system device, all devices specified must be of the same type. For example, if there are two system devices and the equipment EST ordinal for one of them specifies DI-2, the equipment EST ordinal for the other one must also specify DI-2.
- If no devices are specified as system devices, the system library resides on the first nonremovable mass storage device.
- If an ASR entry is made for a device with a SYSTEM entry, the ASR entry is ignored.

The format is:

SYSTEM=eq_1,eq_2,...,eq_n.

Parameter	Description
eq_i	EST ordinal of the device to contain a copy of the system library on the deadstart tape; from 1 through 37_8. This range depends upon the value of NMSD (refer to PPCOM Parameters in section 6); the EQ entry must be set to ON status. One or more ordinals can be specified with one SYSTEM entry. The maximum number of system devices allowed depends upon the value of MXSY (refer to COMSMSC Parameters in section 6).

ASR — ALTERNATE SYSTEM LIBRARY DEVICE ASSIGNMENT ENTRY

This entry specifies which mass storage devices are to be alternate system devices. An alternate system device is a mass storage device on which duplicate copies of system routines can be placed by the system, either for faster access than is possible from a system device or because they are frequently used programs. The following restrictions apply.

- The device must be a mass storage device, including extended memory.
- The device cannot be a removable device.
- The device cannot be a system device. If a SYSTEM entry is made after an ASR entry for the same device, the SYSTEM entry supersedes the ASR entry.

The procedure for selecting the records to be placed on the alternate device is in LIBDECK, section 10.

The format is:

$ASR=eq_1,eq_2,\ldots,eq_n.$

Parameter	Description
eq_i	EST ordinal of mass storage device to be used as an alternative system device; from 1 through 37_8. This range depends upon the value of NMSD (refer to PPCOM Parameters in section 6).

DOWN — LOGICAL DEVICE REMOVAL ENTRY

The DOWN entry specifies which EST ordinals will be logically removed (or down) at deadstart time. You can use this entry for all types of equipment that are defined as nonshared and removable. It is particularly useful either in allowing on-line diagnostics of nonshared extended memory or in removing mass storage devices that cannot be removed on line (perhaps due to hardware malfunctioning).

The format is:

$DOWN=eq_1,eq_2,\ldots,eq_n.$

Parameter	Description
eq_i	EST ordinal of the device to be logically removed; from 1 through 37_8. This range depends upon the value of NMSD (refer to PPCOM Parameters in section 6).

The DOWN entry has the following characteristics.

- It can be entered anywhere after the EQ entries for nonmass storage devices.
- For removable, nonshared mass storage devices, it can be entered anywhere after the EQ and REMOVE entries.
- Shared devices (including the link device) cannot be defined as DOWN in an MMF environment.
- It forces down and off status for the EST ordinal, regardless of what was specified in the EQ entry.
- The deadstart device cannot be defined as DOWN at deadstart time.
- If extended memory is not shared, it can be specified as DOWN at deadstart time, and it will remain DOWN until another level 0 deadstart is performed. Extended memory cannot be brought up on line.

DOWN entry examples:

```
EQ04=DI-1,ON,0,3,32,26.
REMOVE=4.
DOWN=4.

EQ25=CR,OFF,4,,12.
EQ26-LP,ON,6,,12.
DOWN=25,26.

EQ11=DE,OFF,400.
DOWN=11.
```

FAMILY — DEFAULT FAMILY NAME ASSIGNMENT ENTRY

The FAMILY entry defines the default family. The family that is to be defined as the default family may reside on more than one device. The EST ordinal of any device within the family can be specified on the FAMILY entry, except in the following situation. If the member of the family whose device mask will have bit 2^7 (200_8 in mask) set is being initialized, the FAMILY entry must specify the ordinal of this device. In all cases, the FAMILY entry must follow the EQ entry for the device specified.

The following restrictions apply.

- The status parameter for a family device's EQ entry cannot be set to OFF.
- A REMOVE entry cannot exist in the CMRDECK for a device being specified as a FAMILY device.

The format is:

FAMILY=eq.

Parameter	Description
eq	EST ordinal number of the mass storage device that the system automatically uses to determine the user's family when the user does not specify a family name at login or job initiation; from 1 through 37_8. This range depends upon the value of NMSD (refer to PPCOM Parameters in section 6).

REMOVE — REMOVABLE DEVICE ASSIGNMENT ENTRY

If a mass storage device is to be considered removable, you must specify it as such at deadstart with the REMOVE entry. This allows it to be introduced or removed during system operation. A device specified as removable cannot also have an ASR, SYSTEM, TEMP, LINK, FAMILY, DAYFILE, ACCOUNT, ERRLOG, or MAINLOG entry (refer to Central Memory Descriptions in this section for the last four entries) associated with it.

The format is:

REMOVE=eq_1,eq_2,...,eq_n.

Parameter	Description
eq_i	EST ordinal of mass storage device that is to be removable; from 1 through 37_8. This range depends upon the value of NMSD (refer to PPCOM Parameters in section 6). One or more ordinals may be specified with one REMOVE entry.

INITIALIZE — INITIALIZATION ENTRY

To use a mass storage device that is defined with an EQ entry, it must have a label. A label is written on a device when you initialize it by using either the INITIALIZE command, during system operation, or the INITIALIZE entry in the CMRDECK, when it is displayed at the system console at deadstart time (refer to the NOS 2 Operator/Analyst Handbook).

A mass storage device's label is contained on a logical track (usually track 0). It contains information about the allocation and characteristics of a device (and its units, if there is more than one unit on a device). This information is in the form of a label sector for the first unit, a TRT for the device, and a label sector for each unit.

Initialization does not automatically occur at each deadstart because mass storage device labels are recovered during all deadstarts. Therefore, initialize a device only in the following situations.

- To add a new mass storage device (no label exists on the device) use the INITIALIZE entry.
- If parts of the label on a permanent file device have been destroyed by maintenance operations (permanent files having been dumped to another device before diagnostics were run), use the INITIALIZE entry during deadstart to write a new label. Then reload the permanent files.
- If a device (usually a private auxiliary, public auxiliary, or alternate permanent file family device) is added to a system during operation, use the INITIALIZE command (refer to the NOS 2 Operator/Analyst Handbook) to initialize it if it does not have a valid label on it when it is added to the system.
- When an extended memory device is initially placed in maintenance mode, all mainframes using extended memory must initialize it (the maintenance mode parameter is described under EQ - Extended Memory EST Entry). You must also enter the PRESET entry for multimainframe operation.

During a deadstart, the INITIALIZE entry has the following characteristics.

- During a level 0 deadstart, it can be entered at the system console only when the CMRDECK is displayed. It can be entered anywhere after the EQ entry for the device.

  If it is placed in the deadstart tape CMRDECK, the system issues the error message ILLEGAL ENTRY when the CMRDECK is read from the tape.
- A total initialization (op=AL) assumes that no valuable information exists on the device and creates a new label. When the new label is created, all previously existing information on the device, except CTI, CDA, HIVS, and MSL, is lost.
- If the EQ status for the device is OFF when INITIALIZE is entered, initialization of the device occurs whenever the device is set to ON status by the operator with the DSD ON command during normal system operation.
- If the device is not a master device, INITIALIZE (op=AL) only writes a label; if it is a master device, then it also initializes the catalog track and writes EOIs at the beginning of the permit track, the indirect access track (data chain), and each catalog track.
- During a deadstart initialization (op=AL), all flaw reservations specified for a device are lost and must be reentered, except for 844 type devices with factory-formatted disk packs (refer to section 8).

The format is:

INITIALIZE,op,eq_1,eq_2,...,eq_n.

Parameter	Description
op	Level of initialization; one of the following values.

op	Description
AL	Total initialization.
~~HT~~	~~Total initialization as half-track device.~~
~~FT~~	~~Total initialization as full-track device.~~
PF	Initialize permanent files.
QF	Initialize queued files.
DF	Initialize system dayfile.
AF	Initialize account dayfile.
EF	Initialize error log dayfile.
MF	Initialize binary maintenance log dayfile.
FP	Initialize format pack (an automatic selection of AL also occurs).

Parameter	Description
eq_i	EST ordinal of mass storage device to be initialized; from 1 through 37_8. This range depends upon the value of NMSD (refer to PPCOM Parameters in section 6). If the ordinal refers to a family permanent file device, then family name, device number, and mask (if it is a master device) are specified on the PF entry. If it is an auxiliary device, the pack name is specified on the PF entry.

Total initialization (op=AL or FP) is the only initialization that is independent of the content of the pack, if the initialization occurs during deadstart. If the initialization is done while the system is running (refer to the NOS 2 Operator/Analyst Handbook), it is applied to the device after the check mass storage (CMS) routine has recovered it. If CMS cannot recover the device, the initialization is similar to a deadstart initialization (that is, all information on the device is lost).

The device number, family name, and device masks can only be changed during a total initialization. Since all devices may contain permanent files, you should include a PF entry for a device when performing a total initialization. If you do not, the device is assigned a default family name, device number, and device masks. It is possible that these parameters may conflict with other devices in the system. If a conflict occurs, resolve it through the use of PF entries. If you do not specify a PF entry when initializing a device, the default family name is SYSTEM, and the default device numbers begin at 1 and increase by 1 for each device that you initialize without a PF entry. If you initialize equipment 1 without a PF entry, the device mask and secondary mask are set to 377_8. For all other equipment, the default masks are set to 0.

The INITIALIZE entry operates in conjunction with the dayfile entries DAYFILE, ACCOUNT, ERRLOG, and MAINLOG (refer to Central Memory Description in this section) to determine where the dayfiles actually reside. The following examples illustrate the various cases. Assume that the system has three mass storage devices (equipments 1, 2, and 3).

Example 1:

For this example, no dayfile entries are made, and no previous dayfiles exist.

The following CMRDECK entry is made.

```
INITIALIZE,AL,1,2,3.
```

All dayfiles reside on equipment 1.

Example 2:

In this example dayfile entries are made, but no previous dayfiles exist.

The following CMRDECK entries are made.

```
DAYFILE=1,200.
ACCOUNT=2,200.
ERRLOG=3.
MAINLOG=3,200.
INITIALIZE,AL,1,2,3.
```

In this case, the dayfiles reside on the indicated devices (dayfile on equipment 1, account file on equipment 2, error log and binary maintenance log on equipment 3). The default buffer length is used for the error log dayfile buffer.

Example 3:

In this example, dayfile entries are made, and previous dayfiles do exist.

Assume that the CMRDECK entries in example 2 are used.

Since a total initialization has been done on each device, no dayfiles are recovered. They reside on the indicated devices.

Example 4:

In this example, dayfile entries are made, previous dayfiles exist, but no dayfile initialization entries are made.

The following CMRDECK entries are made.

```
DAYFILE=1.
ACCOUNT=2.
ERRLOG=3.
INITIALIZE,PF,1.
```

Two possibilities exist: the dayfiles may already reside on the specified devices, or they may reside on some combination of the possible devices. In either case, since no dayfile initialization entries are made, the old dayfiles are recovered. The residence of these dayfiles is governed by the residence of the old dayfiles. The PF initialization entry returns all permanent file space and relabels the device based on the recovered device parameters. The dayfiles and queued files on this device are not affected by this entry.

Example 5:

In this example, dayfile entries are made, previous dayfiles exist, no dayfile initialization entries are made, and duplicate dayfiles are in existence.

Assume that the CMRDECK entries in example 4 are used.

For the dayfiles that do not have duplicates, the residence is defined by the current residence of the files, not the CMRDECK entries. But assume that an error log dayfile is recovered from equipments 1 and 3. In this case, the most recent file becomes the active error log. Its previous residence overrides the CMRDECK entry. The other file becomes an inactive dayfile (an entry exists in the mass storage table of the device pointing to the inactive file, but the file is not in use by the system).

To produce an inactive error log dayfile, the site must run in the following manner.

1. Assume an 844 disk subsystem with two or more spindles is being used. Run with unit 1 equated to EQ1 and unit 0 unused.

2. Redeadstart, equate unit 0 to EQ1, and do not use unit 1.

3. Redeadstart, equate unit 0 to EQ1, and unit 1 to EQ2.

Since unit 0 has the most recent copy of the error log dayfile, this copy would become an active error log dayfile and the copy on unit 1 would become an inactive error log dayfile.

Example 6:

In this example, dayfile entries are made, the previous dayfiles from example 2 exist, and initialization entries are made.

The following CMRDECK entries are made.

```
DAYFILE=2.
ACCOUNT=2.
ERRLOG=3,300.
MAINLOG=3.
INITIALIZE,DF,1.
INITIALIZE,QF,1.
```

In this case, the account dayfile is recovered and continued on equipment 2. The binary maintenance log dayfile is recovered and continued on equipment 3 with a CM buffer length of zero. The error log dayfile is recovered and continued on equipment 3 with a CM buffer of 300_8 words. The dayfile space on equipment 1 (from example 2) is released and the new dayfile starts on equipment 2. The QF initialization entry releases all space reserved by queued files on equipment 1.

The CM buffer length is not affected by dayfile recovery. It is always specified by the values defined in the CMRDECK entries. If no buffer length entries exist, the system default values are used. The system default value for the CM buffer length of the binary maintenance log dayfile is zero.

LINK — LINK DEVICE DECLARATION

This entry indicates deadstart into a shared multimainframe complex and specifies the equipment number of the link device. The link device must be ECS or ESM, must be identified as either DE or DP, and cannot be defined as removable. It is assigned shared device status automatically. The LINK entry must precede the SHARE entry in the CMRDECK.

The format is:

```
LINK=eq.
```

Parameter	Description
eq	EST ordinal of the ECS or ESM entry; from 1 through 37_8. This range depends upon the value of NMSD (refer to PPCOM Parameters in section 6).

SHARE — SHARED DEVICE ENTRY

This entry identifies the rotating mass storage devices to be shared through a link device (ECS or ESM) by from two through four mainframes in a multimainframe complex. The tables necessary for the management of these devices (MST, TRT, MRT, and DAT) are maintained on the link device. The link device is considered a shared device and need not be specified in the SHARE entry. If the SHARE entry is specified, the ISHARE entry must be omitted.

Following is a list of the equipment types that can be shared.

Equipment	Type
Extended memory	DE
ECS or ESM with DDP	DP
844-21	DI/DK
844-41/44	DJ/DL
885-11/12	DM/DQ

The presence of the SHARE entry implies a multimainframe complex, and as such, a LINK entry is required and must precede the SHARE entry (refer to PRESET - Preset the Link Device Entry and LINK - Link Device Entry in this section).

The format is:

SHARE=eq_1,eq_2,...,eq_n.

Parameter	Description
eq_i	EST ordinal of the mass storage device being shared; from 1 through 37_8. This range depends upon the value of NMSD (refer to PPCOM Parameters in section 6).

Refer to Multimainframe Operation in the NOS 2 System Maintenance Reference Manual for suggestions on shared device configurations.

PRESET — PRESET THE LINK DEVICE ENTRY

This entry defines allocation space and initializes the tables (MST, TRT, MRT, and DAT) on the link device that are required for the management of shared multimainframe mass storage devices. The entry is valid only for level 0 deadstarts by the first mainframe in the multimainframe complex to deadstart. You cannot specify PRESET in the deadstart file CMRDECK; the operator must add it to the CMRDECK when it is displayed during deadstart.

Once PRESET is issued, the SHARE entry is disabled. Therefore, all SHARE entries must precede the PRESET entry. The PRESET entry has two formats:

```
PRESET.
  or
PRESET,n.
```

Parameter	Description
n	Number of shared devices; from 1 through 77_8.

If the operator specifies n, space is allocated for the specified number of shared devices. Use this entry when the total number of shared devices is greater than the number of shared devices defined in the CMRDECK of the first mainframe in the multimainframe complex to do a level 0 deadstart.

If the operator does not specify n, the link device is preset, and the amount of table space reserved for the shared devices is determined by the number of shared entries in the CMRDECK.

NOTE

The PRESET entry for independently shared mass storage devices is described later in this section.

ISHARE — INDEPENDENT SHARED DEVICE DESIGNATION ENTRY

This entry identifies the rotating mass storage devices that are to be independently shared by from 2 through 16 mainframes in a multimainframe complex. The tables necessary for the management of these devices (MST, TRT, MRT, and DIT) are maintained on the shared device. You cannot designate ECS or ESM in an ISHARE entry. When the ISHARE entry is specified, the LINK and SHARE entries must be omitted. Refer to PRESET - Preset the Independent Shared Device Entry in this section.

The following is a list of the equipment types that can be independent shared devices.

Equipment	Type
844-21	DI/DK
844-41/44	DJ/DL
885-11/12	DM/DQ

The format is:

ISHARE=eq_1,eq_2,...,eq_n

Parameter	Description
eq_i	EST ordinal of the mass storage device shared; from 1 through 37_8. This range depends upon the value of NMSD (refer to PPCOM Parameters in section 6).

Refer to Multimainframe Operation in the NOS 2 System Maintenance Reference Manual for suggestions on shared device configurations.

PRESET — PRESET THE INDEPENDENT SHARED DEVICE ENTRY

This entry presets the independent shared devices in a multimainframe complex. The MST, TRT, MRT, and DIT are maintained on the mass storage device itself and are not affected by the PRESET entry. This entry is used in conjunction with the ISHARE entry. It is entered by the operator to the CMRDECK; you cannot specify it in the CMRDECK on the deadstart tape. It is valid only on a level zero deadstart by the first mainframe in the multimainframe complex to deadstart. All ISHARE entries must precede the PRESET entry.

The format is:

PRESET=eq_1,eq_2,...,eq_n.

Parameter	Description
eq_i	EST ordinal of the ISHARE device; from 1 through 37_8. This range depends upon the value of NMSD (refer to PPCOM parameters in section 6).

Refer to Multimainframe Operation in the NOS 2 System Maintenance Reference Manual for suggestions on shared device configurations.

LBC — LOAD BUFFER CONTROLLERS ENTRY

This entry identifies the type of controlware to be installed on the specified disk channels. Depending on the specified parameters, this entry can identify the channels as having half-track or full-track controlware but not install the controlware.

Unless you specify the LBC entry, the system examines the mnemonics of the device in the CMRDECK entry and causes the default version of controlware to be installed as follows:

Device Type	Controlware Version Number
DB †	MA722
DI	MA710
DJ	MA710
DK	MA401
DL	MA401
DM	MA721
DQ	MA721

Use the LBC entry to override these defaults. The format is:

LBC,type,c_1,c_2,...,c_n.

Parameter	Description
type	Controlware to be installed; one of the following values.

type	Description
HT	Install controller with half-track (MA710) controlware.
FT	Install controller with full-track (MA401) controlware.
FM	Install controller with full-track (MA721) controlware.
PH	Install 7155-401 controller with MA722 controlware.
NH	Identify channel(s) as having half-track controller, but do not install the controlware.
NF	Identify channel(s) as having full-track controller, but do not install the controlware.
NM	Identify channel(s) as having full-track 7155-1 controller but do not install the controlware.
NP	Identify channel as having 7155-401 controller, but do not install the controlware.
NN	Identify channel(s) as having NADs but do not install the controlware.

† Not applicable for models 815, 825, 835, and 855.

Parameter	Description
c_i	Disk channels; type determines if controlware is installed on these channels.

The controlware version number that can be loaded into the controller types is as follows:

Controller	Controlware Version Number
7054	MA710
7152	MA710, MA401
7154	MA710, MA401
7155-1	MA721
7155-401	MA7822
380-170	MG101

XM-DECLARE EXTENDED MEMORY SPACE FOR BUFFERS OR USER ACCESS

The XM entry reserves space for I/O buffers and causes initialization of tables for user access to extended memory.

NOTE

Except on models 865 and 875, the assignment of user access to extended memory forces jobs using user access to use CPU 0. This prevents CPUMTR from being locked out during large block transfers to or from extended memory on dual-CPU mainframes.

The format is:

XM=id,iob,uec,EM.

Parameter	Description
id	Identifier of the mainframe that will access the reserved extended memory space; id is required. In a multimainframe complex, you must specify XM entries for each mainframe in the complex in all CMRDECKs, enabling the first mainframe to reserve ECS or ESM space for all mainframes in the complex.
iob	Number of words/1000_8 reserved for buffers for DB, DW, or DV devices; from 40_8 through 7777_8. The optimum number of words to reserve is the amount left after the alternate system library and user ECS are taken into account. If no buffered devices are defined, enter 0.
uec	Number of words/1000_8 of extended memory to reserve for user access; maximum value is 7777_8.
EM	For models 865 and 875 only; if you specify EM, user access extended memory is allocated in UEM regardless of the presence of ECS or ESM. If you omit EM, user extended memory is allocated in the device defined in the extended memory EST entry.

APRDECK 8

The auxiliary mass storage parameter deck (APRDECK) is a text record on the deadstart tape that is processed during system initialization. APRDECK entries identify areas of mass storage that are unusable (flawed areas) and prevent the system from accessing them. The system uses the information in the APRDECK entries to build the TRT for each device that resides in CMR and also in the mass storage device label.

You can place up to 10000_8 APRDECKs on the deadstart file. Placing several APRDECKs on the same deadstart file allows you to use the same file to deadstart several configurations.

The first line in an APRDECK is the deck name. The format of the APRDECK name is:

```
APRapr
```

apr is the number identifying the APRDECK; from 0000 through 7777_8.

An APRDECK must have a name and may have flaw entries. The first APRDECK must contain the deck name APR0000 and nothing else. Subsequent APRDECKs must be numbered consecutively and can contain flaw entries.

The released version of the APRDECK contains no entries. You can enter flaws at three different times.

- During deadstart, after entering all CMRDECK modifications.
- During system operation, using the FLAW entry (refer to the NOS 2 System Maintenance Reference Manual).
- During the configuration of a deadstart file.

If during deadstart, you initialize a device and then enter NEXT, the system displays both the parameters on the device's EST entry and the APRDECK referenced by the EST entry. You can then change the flaws for the device. If the first APRDECK is referenced by the EST entry, the system displays the parameters on the device's EST entry and the APRDECK name, APR0000. You can than enter flaws for the device. These changes to the APRDECKs remain in effect until the next deadstart.

For example, in a CMRDECK, the EST entry for an 844-21 disk is:

```
EQ07=DI-1,ON,0,2,31,33,5.
```

After you initialize equipment 07 and enter NEXT, the following display appears.

```
EQ     TYPE    ST     EQ     UNITS    CHANNELS

07     DI-1    ON     0       02      31  33

APR0005
STK=4173.
STK=7062.
```

The formats described in this section are those for entering flaws during deadstart or on the deadstart file.

- Use the CTK entry to clear all flaw reservations on a device.
- Use the RTK entry to specify the physical address of a flaw in extended memory. (If a reservation for that physical address already exists, it remains in effect.)
- Use the RTK (or TKF) entry to specify the cylinder, track, and sector of a flaw in an 819 disk. (If a reservation for that physical area already exists, it remains in effect.)
- Use the TTK entry to cancel a particular RTK entry. (If that RTK entry does not exist, TTK makes a flaw reservation in extended memory.)
- Use the STK entry to specify the logical address of a flaw. (If a reservation already exists for that logical address, it remains in effect.)

Either obtain flaw addresses from a customer engineer or a systems analyst, or run the MST (mass storage test) on the device to determine the bad areas. MST specifies the physical address of flaws. To find the correspondence between logical and physical track number, refer to mass storage data organization in the NOS 2 Systems Programmer's Instant.

The system reads the flaw information recorded on the utility flaw map of an 881/883/885 disk pack during the initialization of 844/885 equipment and reserves the appropriate areas. For multiunit devices, the flaw reservation is the union of all utility flaw maps. This automatic flawing process occurs in addition to any CTK or STK entry. However, you cannot clear areas recorded as flawed on the utility flaw map of an 881/883/885 disk pack with the CTK entry. Refer to the NOS 2 System Maintenance Reference Manual for information on clearing these flaws.

You can list all APRDECKs on the deadstart file by accessing the SYSTEM file with an ASSIGN or COMMON command, then using the T parameter on the CATALOG command. Refer to the NOS 2 Reference Set, Volume 3, for more information concerning these commands.

CTK — CLEAR DEVICE TRACK RESERVATIONS

The CTK entry clears all flaw reservations previously made with RTK, TKF, STK, or TTK entries. This is the only way to cancel flaw reservations made with STK entries. You can also cancel reservations made with an RTK or TTK entry with a duplicate TTK entry.

The format is:

```
CTK.
```

STK — RESERVE LOGICAL AREAS ON ANY MASS STORAGE DEVICE

Use this entry to specify the logical address of a flaw. If the track was previously reserved, that reservation remains in effect.

The format is:

```
STK=track.
```

Parameter	Description
track	Logical track number. track must be within the range for the device as follows:

track	Device
4000_8 - 7147_8	844-41/44 disk (DJ/DL).
4000_8 - 7222_8	885 disk (DB/DM/DQ).
4000_8 - 7154_8	819 disk (DV/DW).
4000_8 - 7620_8	2048K extended memory (DE).

RTK AND TKF — RESERVE PHYSICAL EXTENDED MEMORY TRACK OR AREAS OF 819 DISKS

Two formats exist for the RTK entry. One format prevents the system from using blocks (tracks) of extended memory. The other format and the TKF entry prevent the system from using sectors on 819 disks.

Use the following format to prevent the system from using blocks (tracks) of extended memory.

RTK=Aaddress.

Parameter	Description
Aaddress	One- through six-digit octal logical address in a track of extended memory; track containing the absolute address is reserved. The letter A must precede the address.

Use either the RTK format or the TKF format to prevent the system from using sectors on 819 disks for model 176.

RTK=Ccylinder,Ttrack,Ssector.
or
TKF=Ccylinder,Ttrack,Ssector.

Parameter	Description
Ccylinder	Cylinder number; from 0 through 627_8 for single-density 819 disks, and from 0 through 1462_8 for double-density 819 disks. The letter C must precede the cylinder number.
Ttrack	Track number; from 0 through 11_8. The letter T must precede the track number.
Ssector	Sector number; 0 through 17_8. The letter S must precede the sector number.

TTK — TOGGLE PHYSICAL EXTENDED MEMORY RESERVATION

To cancel a reservation made with an RTK entry, enter the identical information with a TTK entry. If, however, the reservation did not exist before the TTK is entered, the system reserved the specified area in the same way as with an RTK entry. You can cancel a TTK entry with a duplicate TTK entry.

The format is:

```
TTK=Aaddress.
```

Parameter	Description
Aaddress	One- through six-digit octal logical address in a track of extended memory; track containing absolute address is reserved. The letter A must precede the address.

IPRDECK 9

The IPRDECK contains the system installation parameters that determine the system's operation mode. From 1 through 64 IPRDECKs can exist on a deadstart file. The IPD entry in the CMRDECK specifies which IPRDECK to use. If you omit the IPD entry, the system uses the first IPRDECK on the deadstart file.

There are two IPRDECK console displays. The initial display, IPRINST, is an instruction display. It gives a brief description of all valid IPRDECK entries. The second display is the current IPRDECK. Use the right blank key to switch between the two displays. If either display overflows two screens, press the + key to advance the display.

You can modify the IPRDECK by entering the appropriate changes or additions from the console keyboard. Make these entries while either IPRINST or IPRDECK is displayed. Each console entry supersedes the value currently specified in the IPRDECK.

NOTE

The modified IPRDECK remains in effect only until the next deadstart is performed; that is, changes to the IPRDECK are not recovered across deadstart unless you create a new deadstart file to reflect those changes.

You can list all IPRDECKs on your system by accessing the SYSTEM file with an ASSIGN or COMMON command, then using the T parameter on the CATALOG command. Refer to the NOS 2 Reference Set, Volume 3 for more information concerning these commands.

Most of the IPRDECK entries are also valid DSD commands that can be used to make changes during system operation. A DSD command is not retained after any level of recovery deadstart.

An alphabetical listing of IPRDECK entries with page numbers is contained on the inside front cover.

A listing of the released IPRINST is in appendix D.

JOB CONTROL INFORMATION

The QUEUE, SERVICE, and DELAY entries in IPRDECK relate to job control. General information concerning job control follows.

JOB SCHEDULING

Job scheduling is the control of jobs in the input (IN), executing (EX), and output (OT) queues. Scheduling in the input and output queues is based on the priority of a queue entry relative to all queue entries in the system. The priority of a queue entry depends both upon how long the entry has been waiting in the queue and upon the parameters specified on the QUEUE entry in the IPRDECK. The following formula shows how the system computes the priority; all values are octal.

$$p = \frac{(ct - et)}{WF} + LP$$

Variable	Description
p	Priority; LP $\leq$ p $\leq$ UP. UP (a parameter on the QUEUE entry) is the highest priority.
WF	Weighting factor; a parameter on the QUEUE entry.
ct	Current time in seconds.
et	Time in seconds at which the job entered the queue.
LP	Lowest priority; a parameter on the QUEUE entry.

When an input or output queue entry is created, its priority is the lowest priority (LP) for its service class. The queue priority of the queue entry increases as time passes. The rate at which the priority increases depends upon the weighting factor (WF). The larger the weighting factor, the slower the priority increases. (The queue priority of an entry with a WF of 10_8 increases eight times slower than an entry with a WF of 1.) The queue priority increases either until the queue entry is selected for processing or until the queue priority reaches the highest priority (UP). If the queue priority of an entry reaches UP, it remains at UP until the entry is selected for processing. If the queue priority is zero, the job or file is never selected by the job scheduler and stays in the queue until the operator either enters a DROP command or resets the priority to a nonzero number.

Queue control for job execution determines how much continuous execution time a job gets. The amount of execution time depends both on the parameters specified on the QUEUE and SERVICE entries in the IPRDECK and whether other jobs are waiting for execution.

When a job in the input queue is selected for execution, it gets an initial priority (IP; a parameter on the QUEUE entry) for the execution queue. Scheduling priority increases for a job in the execution queue in the same way as in the input and output queues. When the job is selected for execution, it executes either until completion or until it is preempted by a higher priority job. When a job exceeds its first time slice, its priority becomes the initial slice priority (IL; a parameter on the QUEUE entry). On expiration of subsequent time slices the job reenters the execution queue with the lowest priority (LP; a parameter on the QUEUE entry). The IL and LP parameters provide two opportunities for you to change the job's scheduling priority.

For the interactive service class, there is an additional execution queue priority, TP (a parameter on the SERVICE entry). It is assigned to the execution queue entry of a job restarting after terminal I/O. The value of TP aids response time to program prompts. Also, for the interactive service class, the initial priority (IP) has an added significance. In addition to being the priority at which jobs are scheduled from the input queue to the execution queue, IP is the priority assigned to each newly initiated interactive job step. Using separate TP and IP parameters allows the system to give faster responses to users interacting with a job. To achieve this, a value for the TP parameter slightly greater than the value for the IP parameter is recommended (refer to table 9-1 and figure 9-1).

The relative values of the QUEUE and SERVICE parameters, both among service classes and within a service class, affect system performance. For an example of ranges of service class priorities, refer to figure 9-1. For an example set of specific entries for the QUEUE and IPRDECK parameters, refer to table 9-1.

JOB CONTROL

The parameters on the SERVICE entry in the IPRDECK direct the control of the jobs. The parameters set the:

- Initial CPU priority at job initiation.
- CPU time slice in milliseconds divided by 64.
- Central memory time slice in seconds.
- Number of jobs per service class.
- Timeout delay for interactive and detached jobs.

A job leaves a central memory control point because:

- A job completes, aborts, or is suspended.
- A system request causes a job to be rolled out.
- Terminal input/output is required.
- The control point is made available for a higher priority job.

The first category is self-explanatory.

In the second category, a system request includes a job request for a tape or disk pack, a ROLLOUT command from the generator, and execution of the ROLLOUT macro. Whenever a job rolls out, it is assigned whatever priority it has at that point. The priority increases as time passes, giving that job a better chance to be selected for execution again. When the job is selected and rolled in, its priority becomes the value of the UP parameter on the QUEUE entry, and the job scheduling priority cycle begins again. This description on job rollout applies to batch, remote batch, and interactive jobs that are not doing interactive I/O. For interactive jobs that do terminal I/O within a time slice, behavior is slightly different. When I/O is complete and input, for example, is available, the system assigns the rolled out job the terminal I/O scheduling priority (TP parameter on the SERVICE entry). The TP parameter gives the job a priority equal to the priority of jobs still within their initial time slice, an advantage over jobs in a second time slice, and a larger advantage over jobs in a third or higher time slice.

For the third category, a job leaves central memory when:

- The system requests terminal input.
- You request terminal output and the recall parameter is specified on the request.
- You issue a RECALL macro after a request for terminal output that omitted the recall parameter.

The fourth category is the mechanism that ensures reasonable service to all users in the system. The operating system controls the amount of central processor or central memory time each type of job can use when it is at a control point. This ensures that one job does not monopolize system resources.

If a job exceeds either the central processor or central memory time slice, and it is not a subsystem, the scheduling priority is set to the lower bound priority (the IL parameter on the QUEUE entry) for its service class. Thus, any job in the queue with a higher priority forces the executing job with the lower priority to be rolled out. The rolled-out job ages normally until its priority is higher than the priorities of either the jobs in the input queue or a job that is executing; then it is again scheduled to a control point.

Once a job is scheduled, it is desirable to use the resources allocated before another job forces it out. If a job maintained its scheduling priority when it was assigned to a control point, another job could age past that job and force it to be rolled out before it had an opportunity to use its time slice. For this reason, when a job is assigned to a control point and its priority is within the queue aging range, it is given a priority equal to the highest priority (the UP parameter in the QUEUE entry) for its service class.

Selecting the number of control points available on the system depends on the amount of memory space available, the job mix, and the mode in which the system is being run. Each control point needs 200_8 words of CMR space. For example, if an installation is running only TAF, then four or five control points may suffice. On the other hand, if the system is running a large number of interactive terminals with heavy permanent file activity, then 20 or more control points may be needed. Section 7 describes the CMRDECK entry for selecting the number of control points. You may need to study memory and control point use in order to correctly determine the setting of this option. If memory use is high and control point use is low, select fewer control points. If control point use is high and memory use is low, select more control points.

MEMORY CONTROL

You can control the maximum memory allowed for job types and for service classes with the parameters you specify on the SERVICE entry in the IPRDECK.

These parameters specify:

- Maximum field length divided by 100_8 for a job in a service class.
- Maximum field length divided by 100_8 for all jobs of the specified service class.
- Maximum extended memory length in words divided by 1000_8 for a job in a service class.
- Maximum extended memory length in words divided by 1000_8 for all jobs of the specified service class.

Initially, the scheduler attempts to find the highest priority job that meets the memory constraints. However, if the scheduler is unable to schedule a job and has explicitly rejected one or more jobs because of these memory constraints, it attempts to schedule a job a second time. During this second attempt, any job that requires other jobs to be rolled out is not scheduled. Otherwise, the constraints are ignored, and the job is scheduled at the lower bound priority, IL (a parameter on the QUEUE entry). This means that the constraints are applied as long as there are enough jobs of each service class. However, if central memory is unused because batch jobs are at a maximum and no other jobs are available, the scheduler attempts to schedule the batch jobs.

All of these parameters can be changed by using the SERVICE, QUEUE, and DELAY entries.

EXAMPLE OF JOB CONTROL PARAMETERS

An example set of entries for the job control parameters is shown in table 9-1. These entries fall within the example of ranges of service class priorities shown in figure 9-1. Neither the specific entries nor the ranges are recommended; they are strictly examples to aid you in selecting QUEUE and SERVICE entry parameters.

The following discussion indicates the significance of the values chosen and how they relate to each other.

The entry (lowest) priority (LP parameter in the QUEUE entry) of the system service class input queue is higher than all entry priorities, except the network supervision and subsystem entry priorities, because it is assumed that an operator-initiated job should receive prompt attention. A system job rolls out any batch job. Most system jobs and all subsystems are coded to adjust their priorities correctly, once execution begins. The entry priority (LP) of the network supervision service class input queue is set high to ensure adequate response time from network programs and facilities such as CS, NS, and NVF.

The queue priorities for batch, remote batch, and detached jobs are similar. The time slice for detached jobs is shorter than for the batch and remote batch jobs. The assumption is that detached jobs need less CPU time than either batch or remote batch jobs.

The queue priorities are explained in Job Scheduling, in this section.

The time slices for the various service classes reflect the following objectives.

- To keep system jobs with their high entry priority from monopolizing system resources.
- To keep at a minimum rollout activity caused by diagnostics running as maintenance service class jobs.
- To allow most interactive jobs to compile, load, and begin execution in one time slice.
- To give batch jobs a large time slice, because little is gained from rolling out batch jobs. There is no problem with the time slices for batch jobs compared to interactive jobs, because, with the priorities shown, an interactive job generally causes a batch job to roll out.
- To ensure prompt service to all interactive users, without employing an excessive number of rollouts, by setting the time slices for interactive jobs low. The time slice parameters are critical to good interactive performance. In some cases, depending on the system load, job size, and so forth, it may be desirable to change these parameters during operation.

Interactive performance is quite sensitive to the relative values of the QUEUE and SERVICE scheduling parameters both within a service class and among service classes. Table 9-1 shows almost no overlaps of values except that batch, remote batch, and detached jobs could age slightly past interactive jobs if they remained rolled out for a very long time (about 26 minutes). For the interactive service class, the range between entry (lowest) priorities and the highest priority is wide, so that few jobs are at the highest priority simultaneously. If many jobs reach the highest priority, their priorities are the same, and the order in which the jobs entered the queue is lost. The terminal I/O priority (TP) is set slightly higher than the initial priority (IP) to reduce response time for the user interacting with a job as compared to the user initiating a new job step. This parameter setting improves the perceived responsiveness of the system for the interactive users.

Table 9-1. Example Set of Job Control Parameters[†]

	Input Queue QUEUE Parameters			Execution Queue QUEUE Parameters					
Service Class	LP	UP	WF	LP	UP	WF	IP	IL	TP
SY (system)	7770	7776	1	2000	7000	1	7000	4000	-
BC (batch)	10	4000	1	1000	4004	1	2000	2000	-
RB (remote batch)	10	4000	1	1000	4004	1	2000	2000	-
CT (communication task)	7770	7776	1	3000	7000	1	7000	4000	-
TS (interactive)	7000	7770	1	3700	7000	1	4004	3770	4024
NS (network supervision)	7770	7776	1	7770	7776	1	7772	7772	-
DI (detached)	10	4000	1	1000	4000	1	2000	2000	-
SS (subsystem)	7770	7776	1	7770	7776	1	7772	7772	-
MA (maintenance)	1	10	1	1	10	1	10	4	-

	Output Queue QUEUE Parameters			Time Slice SERVICE Parameters		CPU Priority PR SERVICE Parameter	Number of Jobs NJ SERVICE Parameter	Timeout Delay TD SERVICE Parameter
Service Class	LP	UP	WF	CP	CM			
SY (system)	7000	7776	1	100	20	30	7777	-
BC (batch)	1	7000	1	400	200	30	7777	-
RB (remote batch)	1	7000	1	400	200	30	7777	-
CT (communication task)	7000	7776	1	400	200	30	7777	-
TS (interactive)	1	7000	1	40	10	30	7777	113[††]
NS (network supervision)	1	7000	1	400	200	31	7777	-
DI (detached)	1	7000	1	100	20	30	7777	341[†††]
SS (subsystem)	7400	7776	1	100	20	30	7777	-
MA (maintenance)	7000	7776	1	100	20	2	7777	-

[†]All values are octal, and the DELAY parameters are:
JS=1, CR=30, AR=1750, MN=10, MX=20, and JQ=2.
[††]A TS suspended job times out after 10 minutes when the timeout delay is 113.
[†††]A DI suspended job times out after 30 minutes when the timeout delay is 341.

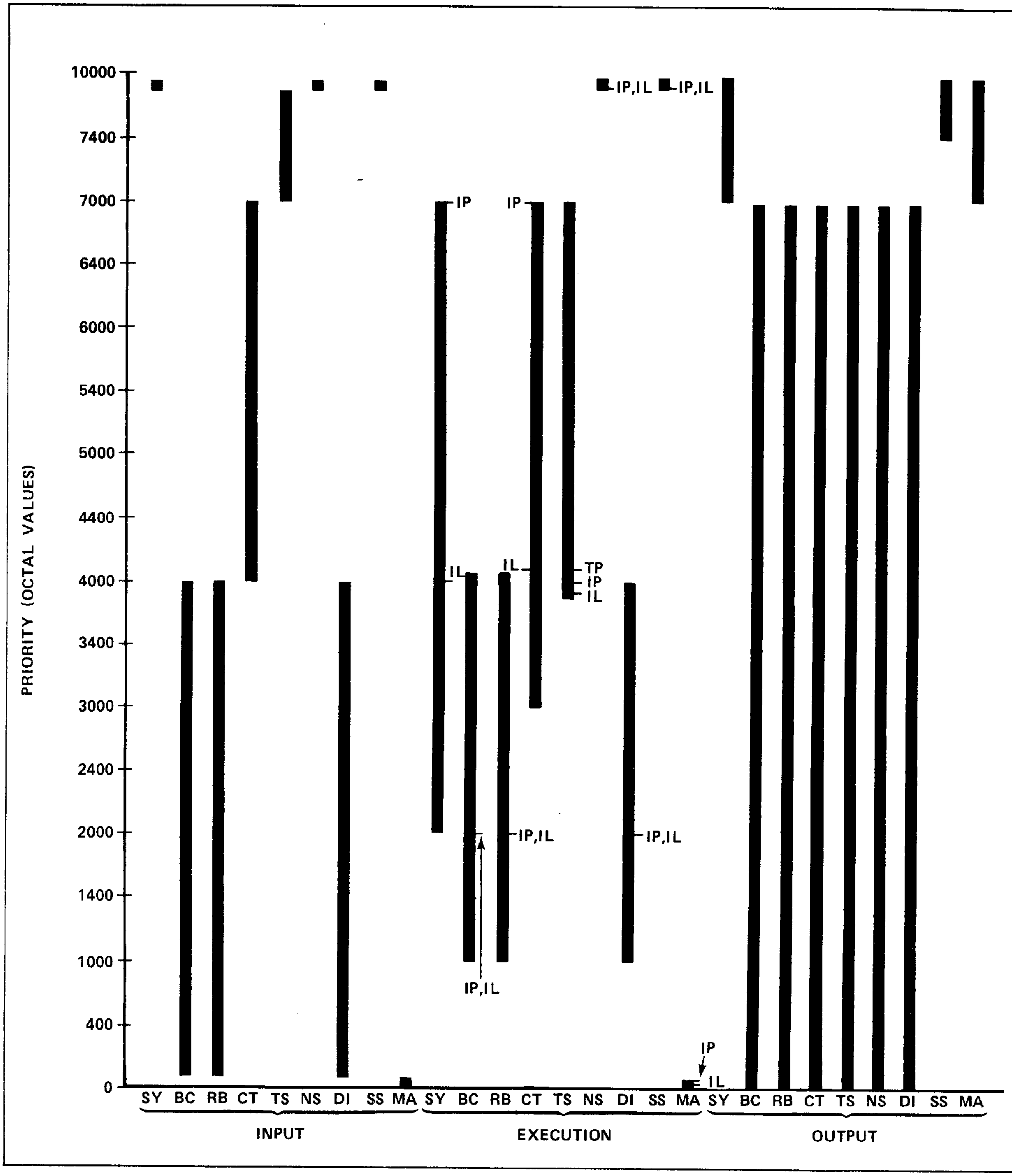

Figure 9-1. Example of Ranges of Service Class Priorities

The CPU priorities reflect the following objectives.

- The maintenance service class jobs are run at the lowest priority. This handles the background CPU and memory diagnostics.

- All other jobs, except network supervision service class jobs, run at the same priority. It is generally not desirable to run one class of jobs at a higher priority than another because the system would roll in jobs that occupy memory without executing until they exceed the central memory time slice.

- The network supervision service class is set high to ensure adequate performance from network programs and facilities such as CS, NS, and NVF.

ENTRIES MADE ONLY DURING DEADSTART

You can enter the following IPRDECK entries on the IPRDECK only during deadstart. They cannot be entered as DSD commands. Changes to the IPRDECK are not retained across deadstart unless a new deadstart tape is created to reflect those changes. The entries are described in alphabetical order.

CPM

Format:

$CPM,s_1=n_1,s_2=n_2.$

Default:

System selection.

Significance:

This entry alters the central processor multiplier of type s_i, which is used in SRU calculations. The s_i parameters are either 0 or 1 to indicate the multipliers S0 or S1, respectively. Entering 0=n obtains a multiplier to be used for S0 and entering 1=n obtains a multiplier to be used for S1. (Refer to the NOS 2 System Maintenance Reference Manual for a discussion of multiplier use.) The values of n_i range from 1 through 40_8 and are used as indexes to values defined in COMSSRU in order to determine the multiplier value. The default values are listed in table 9-2.

Table 9-2. CPM Default Values for n_i (Sheet 1 of 2)

n_i (Octal)	COMSSRU Name	Mainframe Model	COMSSRU Default Multiplier Value
1	CP62	6200	1.0
2	CP64	6400	1.0
3	CP65	6500	1.0
4	CP66	6600	1.0
5	CP67	6700	1.0
6	CP71	71	1.0
7	CP72	72	1.0

Table 9-2. CPM Default Values for n_i (Sheet 2 of 2)

n_i (Octal)	COMSSRU Name	Mainframe Model	COMSSRU Default Multiplier Value
10	CP73	73	1.0
11	CP74	74	1.0
12	C171	171	1.0
13	C172	172	1.0
14	C173	173	1.0
15	C174	174	1.0
16	C175	175	1.0
17	C176	176	1.0
20	C720	720	1.0
21	C730	730	1.0
22	C740	740	1.0
23	C750	750	1.0
24	C760	760	1.0
25	C815	815	1.0
26	C825	825	1.0
27	C835	835	1.0
30	C855	855	1.0
31	C865	865	1.0
32	C875	875	1.0
33 through 40	Reserved for Control Data		
41	ICM1	Model on which you are installing.	1.0
42	ICM2	Model on which you are installing.	2.0
43	ICM3	Model on which you are installing.	3.0
44	ICM4	Model on which you are installing.	4.0
45	ICM5	Model on which you are installing.	5.0

CSM

Format:

CSM=csm.

Default:

64

Significance:

This entry sets the operating system character set mode. To change the character set mode for the products, a change must be made in IPARAMS, and the products must be reassembled.

csm	Description
63	63-character set.
64	64-character set.

The system assumes a 64-character set if there is no CSM entry in the current IPRDECK.

NOTE

Unpredictable and possibly serious problems occur if the operating system is operating in one character set and the products are operating in another. Therefore, ensure that all installed products and the operating system are in the same mode.

DSD

Format:

DSD,level,$command_1$#$command_2$#...#$command_n$

Default:

None.

Significance:

This entry specifies the initial command(s) to be executed by the DSD program when the deadstart is complete.

Parameter	Description
level	Level of deadstart (0, 1, 2, or 3).
$command_i$	DSD command to be executed for the level of deadstart specified. Several commands can be specified by separating them with the # (6-bit display code 60) or % (6-bit display code 63) character. These characters may misposition parts of the console display of IPRDECK if they appear as the upper 6 bits in a byte.

Example:

```
DSD,0,MAI%X.QREC(PO=N)
```

EXTENDED STACK PURGING

Formats:

```
ENABLE,EXTENDED STACK PURGING.
DISABLE,EXTENDED STACK PURGING.
```

Default:

Disabled.

Significance:

These entries specify the default action for instruction-stack purging for nonsystem-origin jobs on models 815, 825, 835, and 855. Refer to the MODE macro in the NOS 2 Reference Set, Volume 4, for a description of instruction-stack purging.

KEYPM

Format:

```
KEYPM=cc.
```

Default:

26

Significance:

This entry specifies the keypunch mode to be assumed during system operation.

cc	Description
26	026 keypunch mode.
29	029 keypunch mode.

This entry is used for all batch jobs submitted if the keypunch mode is not specified on the job command. This does not apply to RBF.

LID

Format:

LID=lid,a_1,a_2,a_3,a_4.

Default:

No attributes.

Significance:

This entry specifies a mainframe logical identifier and the RHF attributes for that mainframe.

Parameter	Description
lid	Three alphanumeric character identifier to be placed in the LID table.
a_i	Any combination of the following attributes to be specified for the mainframe. H lid represents a host mainframe. L lid represents a linked (remote) mainframe. V User validation is required. D lid is disabled. Host lids always require user validation; therefore, H specified without V is invalid.

NOTE

A blank LID entry terminates the LID table. A lid with no attributes indicates that the lid is not valid but it does not terminate the LID table.

PROBE

Formats:

ENABLE,PROBE.
DISABLE,PROBE.

Default:

Disabled.

Significance:

These entries enable and disable the data gathering facility of CPUMTR.

SCP

Formats:

ENABLE,SCP.
DISABLE,SCP.

Default:

Disabled.

Significance:

These entries specify whether to use the system control point facility. You must enable SCP if CDCS, MSS, NAM, or TAF will be used. If none of these will be used, leave SCP disabled so that more CMR space is available.

SCRSIM

Formats:

ENABLE,SCRSIM.
DISABLE,SCRSIM.

Default:

Disabled.

Significance:

These entries enable or disable the simulation of the status/control register via the interlock register on CYBER 70 Computer Systems (refer to the NOS 2 System Maintenance Reference Manual for information on the SCRSIM utility).

SUBCP

Formats:

ENABLE,SUBCP.
DISABLE,SUBCP.

Default:

Disabled.

Significance:

These entries specify whether CPUMTR is to be initialized to handle subcontrol point (TAF) processing.

If SUBCP is disabled, CPUMTR is not initialized to handle subcontrol point processing. If you are not running TAF and if no user applications use subcontrol point processing, disable SUBCP so that CPUMTR uses less CMR space.

TCVM

Format:

TCVM=mode.

Default:

AS

Significance:

This entry sets the tape conversion mode to be assumed during system operation.

mode	Description
AS	ASCII nine-track conversion.
US	ANSI (previously known as USASI) nine-track conversion (same as AS).
EB	EBCDIC nine-track conversion.

TDEN

Format:

TDEN=density.

Default:

HY for seven-track tapes.
PE for nine-track tapes.

Significance:

This entry sets the system tape density. When the density is set, any tape unit accessed is automatically set to this density unless specified otherwise by a magnetic tape request. Two TDEN entries may be present, one for seven-track and one for nine-track.

density	Description
LO	200 cpi (seven-track).
HI	556 cpi (seven-track).
HY	800 cpi (seven-track).
HD	800 cpi (nine-track).
PE	1600 cpi (nine-track).
GE	6250 cpi (nine-track).

TDTR

Format:

TDTR=tracktype.

Default:

NT

Significance:

This entry sets the default track type.

tracktype	Description
MT	Seven-track.
NT	Nine-track.

DEADSTART ENTRIES AND DSD COMMANDS

You can enter the following IPRDECK entries on the IPRDECK both during deadstart and as DSD commands during system operation. The reason for entering them as DSD commands is to change the system's operation between deadstarts. DSD changes are not retained across deadstarts. IPRDECK changes are not retained across deadstarts unless a new deadstart tape is created to reflect those changes.

A description of the enabling and disabling of subsystems follows immediately. Then the remaining IPRDECK entries are described in alphabetical order.

SUBSYSTEMS

Subsystems are enabled and disabled with the IPRDECK entries:

ENABLE,subsystem,cp.
DISABLE,subsystem,cp.

Parameter	Description
subsystem	Three characters that select the desired subsystem; one of the following values.

subsystem	Description
BIO	Central site batch I/O for line printers, card readers, and card punches.
CDC	CYBER Database Control System.
IAF	Interactive Facility. Do not specify the cp parameter on the ENABLE entry for IAF.

Parameter	Description

subsystem	Description
MAG	Magnetic tape subsystem. Enable MAG if removable auxiliary packs are used. Disabling MAG frees a control point for other use.
MAP	MAP III.
MCS	Message Control System.
MSS	Mass Storage Subsystem.
NAM	Network Access Method.
RBF	Remote Batch Facility.
RDF	Remote Diagnostic Facility.
RHF	Remote Host Facility.
STM	STIMULA. Do not specify the cp parameter on the ENABLE entry for STM.
TAF	Transaction Facility.

cp	Control point where the subsystem will reside. If you omit cp, the current control point is used. If you enter zero as the control point, the subsystem may reside at any control point. Omit cp for IAF and STM.

If you make no entry in the IPRDECK for a subsystem, the subsystem is disabled, except for BIO and MAG, which are enabled.

DEBUG

Format:

DEBUG.

Default:

Disabled.

Significance:

This entry selects or clears debug mode, depending upon the current status.

If enabled, debug mode is selected. The message DEBUG appears in the header of the left screen display. Debug mode provides system origin privileges to validated users and allows modifications to be made to the running system.

If disabled, debug mode is cleared. Control Data recommends not allowing debug mode in a normal production environment.

DELAY

Format:

DELAY,CRcr,ARar,MXmx,MNmn,JQjq,JSjs.

Default:

None.

Significance:

This entry specifies the system delay parameters. Refer to table 9-1 for an example set of parameter entries. Refer to figure 9-1 for an example of ranges of service class priorities.

Parameter	Description
CRcr	CPU recall delay in milliseconds; cr is from 1 through 7777_8.
ARar	PP/auto recall delay in milliseconds; ar is from 1 through 7777_8.
MXmx	Maximum job switch delay in milliseconds; mx is from 1 through 7777_8.
MNmn	Minimum job switch delay in milliseconds; mn is from 1 through 7777_8.
JQjq	Exponent used to determine the input job (QFT to EJT) scheduling delay; jq is from 0 through 14_8 seconds. The delay in seconds between the scheduling of input jobs is calculated as follows: $\text{delay} = 2^{jq}$
JSjs	Job scheduler delay in seconds; js is from 1 through 7777_8.

ENGR

Formats:

ENABLE,ENGR.
DISABLE,ENGR.

Default:

Disabled.

Significance:

These entries enable or disable engineering mode. If enabled, the ENGR message appears in the header of the left screen display. Engineering mode allows the PPU/hardware diagnostics and the 881/883 pack reformatting utility FORMAT to run while the system is in operation.

FILE STAGING

Formats:

```
ENABLE,FILE STAGING.
DISABLE,FILE STAGING.
```

Default:

Disabled.

Significance:

These entries specify whether permanent files that reside on the Mass Storage Facility (MSF) are staged to disk. If disabled, jobs attempting to access MSF resident files are aborted.

LOCK

Format:

```
LOCK.
```

Default:

Unlocked.

Significance:

This entry specifies the system is locked. This software function prevents entry of restricted commands; all other DSD commands can be entered. Refer to the UNLOCK command in the NOS 2 Operator/Analyst Handbook for the list of restricted commands. The console is normally locked when the system is being used in a production environment.

LOGGING

Formats:

```
ENABLE,LOGGING.
DISABLE,LOGGING.
```

Default:

Disabled.

Significance:

These entries specify whether dayfile messages intended for systems analysts are logged in the dayfile. The dayfile messages are documented in an appendix of the NOS 2 Reference Set, Volume 3.

MASTER MSS

Formats:

```
ENABLE,MASTER MSS.
DISABLE,MASTER MSS.
```

Default:

Disabled.

Significance:

These entries specify whether the MSS executive, when initialized, is to run in master (enabled) or slave (disabled) mode.

MS VALIDATION

Formats:

```
ENABLE,MS VALIDATION.
DISABLE,MS VALIDATION.
```

Default:

Disabled.

Significance:

This entry enables or disables mass storage validation. If enabled, CMR is increased by 60_8 words, and the system verifies that, for each mass storage device, the sum of the counts of unreserved tracks and preserved files equals values specified in the device's mass storage table.

If the device is a master device (contains user catalogs), the system also verifies that:

- The device's track reservation table specifies that the first tracks of the indirect access file chain and the permit area are reserved and preserved.
- The label track is linked to the first catalog track.
- The number of catalog tracks is a power of 2.
- The catalog chain is reserved, of correct length, and contiguous if flagged as such in the device's MST.

In order to enable/disable mass storage validation with a DSD command entry (refer to the NOS 2 Operator/Analyst Handbook), enable it in the IPRDECK during a level 0, 1, or 2 deadstart.

PF VALIDATION

Formats:

```
ENABLE,PF VALIDATION.
DISABLE,PF VALIDATION.
```

Default:

Disabled.

Significance:

These entries enable or disable preserved file (PF) validation. If enabled, the system aborts an attach of a direct access permanent file if its end-of-information was altered during recovery of the file. If NA (no abort) is specified on the attach request, the system attaches the file.

If mass storage validation is also enabled, TRT verification of preserved files takes place during a level 3 deadstart as follows:

- For all files, the system ensures that all tracks are reserved and that no circular linkage exists.
- For all queued, permanent direct-access, and fast-attach files, the system also ensures that the first track is preserved.

If mass storage validation is enabled on a level 1 or 2 deadstart, TRT verification takes place automatically, regardless of the status of PF VALIDATION.

PRIVILEGED RDF

Formats:

```
ENABLE,PRIVILEGED RDF.
DISABLE,PRIVILEGED RDF.
```

Default:

Disabled.

Significance:

These entries enable and disable privileged mode of RDF. If enabled, a user's commands are not checked to ensure that a maintenance function is being performed.

QUEUE

Format:

QUEUE,sc,qt,LPlp,UPup,IPip,WFwf,ILil.

Default:

None.

Significance:

This entry specifies the queue priorities associated with the input, executing, and output queues for each job service class. Refer to table 9-1 for an example set of parameter entries and to figure 9-1 for an example of ranges of service class priorities.

Parameter	Description
sc	Service class.
	<table><tr><th>sc</th><th>Description</th></tr><tr><td>BC</td><td>Batch.</td></tr><tr><td>CT</td><td>Communication task.</td></tr><tr><td>DI</td><td>Detached interactive.</td></tr><tr><td>MA</td><td>Maintenance.</td></tr><tr><td>NS</td><td>Network supervisor.</td></tr><tr><td>RB</td><td>Remote batch.</td></tr><tr><td>SS</td><td>Subsystem.</td></tr><tr><td>SY</td><td>System.</td></tr><tr><td>TS</td><td>Interactive.</td></tr></table>
qt	Job queue type.
	<table><tr><th>qt</th><th>Description</th></tr><tr><td>EX</td><td>Executing jobs.</td></tr><tr><td>IN</td><td>Input queued files.</td></tr><tr><td>OT</td><td>Output queued files.</td></tr></table>
LPlp	Lowest priority at which a file or job can enter the specified queue. lp ranges from 0 through 7777_8.
UPup	Highest priority a job or queued file can reach in the specified queue. up ranges from 0 through 7777_8.

Parameter	Description
IPip	Initial priority only for executing jobs (EX). ip ranges from 0 through 7777_8.
WFwf	Weighting factor for queue priority calculation. wf must be 1, 2, 4, 10_8, 20_8, or 40_8. The smaller the weighting factor, the faster the queue entry reaches its highest priority.
ILil	Lower bound priority for the initial time slice. il ranges from 0 through 7777_8. This parameter is valid only for executing jobs (EX).

REMOVABLE PACKS

Formats:

ENABLE,REMOVABLE PACKS.
DISABLE,REMOVABLE PACKS.

Default:

Enabled.

Significance:

These entries enable or disable automatic label checking for mass storage devices that are defined as removable.

If REMOVABLE PACKS is enabled, automatic label checking occurs. This status must be available to perform label verification before removable devices can be accessed.

If REMOVABLE PACKS is disabled, any removable devices introduced into the system will not be recognized.

RESIDENT RDF

Formats:

ENABLE,RESIDENT RDF.
DISABLE,RESIDENT RDF.

Default:

Disabled.

Significance:

These entries enable and disable resident mode of RDF. While in resident mode, RDF remains active, regardless of terminal inactivity, until RDF is disabled. When resident mode is disabled, RDF becomes inactive if no one is logged on at the remote diagnostic terminal for 15 minutes.

SECONDARY USER CARDS

Formats:

```
ENABLE,SECONDARY USER CARDS.
DISABLE,SECONDARY USER CARDS.
```

Default:

Disabled.

Significance:

The enable option allows jobs to issue more than one USER command. If the option is disabled, any USER command encountered after the first causes the job to abort without EXIT command processing. Also, the security count for the current user name is decreased accordingly.

SERVICE

Format:

```
SERVICE,sc,PRpr,CPcp,CMcm,NJnj,FLfl,AMam,ECec,EMem,FCfc,CScs,FSfs,DSds,TDtd,TPtp.
```

Default:

None.

Significance:

This entry specifies the service limits associated with each service class. Refer to table 9-1 for an example set of parameter entries and to figure 9-1 for an example of ranges of service class priorities.

Parameter	Description
sc	Service class; one of the following values.

sc	Description
BC	Batch.
CT	Communication task.
DI	Detached interactive.
MA	Maintenance.
NS	Network supervisor.
RB	Remote batch.
SS	Subsystem.
SY	System.
TS	Interactive.

Parameter	Description
PRpr	CPU priority; pr ranges from 2 through 70_8. All service classes except network supervisor (NS) and maintenance (MA) jobs are normally set to the same CPU priority. Since jobs with lowest priority access the CPU last, MA jobs are usually set to the lowest priority to prevent them from interfering with other system activity.
CPcp	CPU time slice/100_8 in milliseconds. This parameter specifies the maximum amount of time a job of the specified service class can use the CPU before its scheduling priority is set to its lower bound priority. cp must be an octal number.
CMcm	Central memory time slice in seconds; cm ranges from 0 through 7777_8. This parameter specifies the maximum amount of time a job of the specified service class can remain at a control point before it becomes eligible to be rolled out.
NJnj	Maximum number of jobs; nj ranges from 0 through 7777_8. This parameter specifies the number of jobs that can be active at one time in the specified service class.
FLfl	Maximum field length/100_8 for any job of the specified service class; fl ranges from 0 through 7777_8. Jobs with field length requirements that exceed this value are not considered for scheduling, unless no other jobs are to be run and sufficient unused memory is available to run the job without rolling out a running job. However, an interactive service class job that exceeds its maximum field length aborts. You typically use this parameter to limit the memory requirement for jobs of a specific service class during certain hours of the day. For example, you may use the FL parameter to specify a maximum field length for all batch service class jobs between the hours of 2 and 4 p.m.
AMam	Maximum field length/100_8 for all jobs of the specified service class; am ranges from 0 through 7777_8. This parameter partitions central memory by limiting the field length available to each service class. For example, if a job whose field length exceeds that specified for its service class is scheduled to a control point, it may not be scheduled until the required field length is available. This means that a lower priority job from a different service class may be scheduled first. However, a job that would normally exceed the field length for its service class can be scheduled to a control point if not enough jobs exist to fill the field length specified for another service class. The system attempts to use central memory to its greatest capacity.
ECec	Maximum extended memory length in words divided by 1000_8 for any job of the specified service class; ec ranges from 0 through 7777_8.
EMem	Maximum extended memory length in words divided by 1000_8 for all jobs of the specified service class; em ranges from 0 through 7777_8.

Parameter	Description
FCfc	Number of permanent files allowed. fc indicates a limit value which is the maximum number of permanent files allowed.

fc	Limit Value
0	Unlimited
1	10_8
2	20_8
3	30_8
4	40_8
5	50_8
6	100_8
7	Unlimited

CScs	Cumulative size in PRUs allowed for all indirect access permanent files. cs indicates a limit value for the cumulative size.

cs	Limit Value
0	Unlimited
1	1000_8
2	2000_8
3	5000_8
4	10000_8
5	50000_8
6	100000_8
7	Unlimited

Parameter	Description
FSfs	Size in PRUs allowed for individual indirect access permanent files. fs indicates a limit value for the size of the files (refer to COMSPFM Parameters in section 6).

fs	Limit Value
0	Unlimited
1	10_8
2	30_8
3	50_8
4	100_8
5	150_8
6	300_8
7	Unlimited

DSds Size in PRUs allowed for individual direct access permanent files. ds indicates the limit value for the size of the files.

ds	Limit Value
0	Unlimited
1	1000_8
2	2000_8
3	5000_8
4	10000_8
5	50000_8
6	100000_8
7	Unlimited

TDtd Suspension timeout delay; td ranges from 0 through 7777_8. A suspended job will not be timed out for td x 10_8 seconds. The maximum delay is approximately 9 hours.

TPtp Initial scheduling priority for on-line interactive jobs; tp ranges from 0 through 7777_8.

SRST

Format:

SRST=n.

Default:

0

Significance:

This entry specifies the secondary rollout sector threshold. Any rollout file smaller than n sectors ($0 \leq n \leq 7777_8$) is considered a secondary rollout file for the purpose of equipment selection (refer to MSAL - Mass Storage Allocation Control Entry in section 7).

NOTE

The size of the rollout file for any job must be at least seven sectors larger than the combined size in sectors of the job's central memory and extended memory field lengths.

UNLOCK

Format:

UNLOCK.

Default:

Unlocked.

Significance:

This entry specifies the system console is unlocked. All DSD commands can be entered when the console is unlocked. The console is usually locked when the system is being used in a production environment. Refer to LOCK in this section.

USER ECS

Formats:

ENABLE,USER ECS.
DISABLE,USER ECS.

Default:

Disabled.

Significance:

These entries enable or disable scheduling of jobs that access user extended memory.

LIBDECK 10

LIBDECK is a SYSEDIT directive record on the deadstart tape. SYSEDIT reads LIBDECK during the system load. LIBDECK specifies program residence, field length, record type, and parameter format.

Up to eight LIBDECK records can be placed on the deadstart tape. Additional records are named LIBDCKn, where $1 \leq n \leq 7$. A specific record can be selected with a LIB=n entry in CMRDECK. The multiple LIBDECK (CMRDECK/APRDECK/IPRDECK) capability enables the use of a single deadstart file on virtually any system configuration.

You can list all LIBDECKs on the deadstart file by accessing the SYSTEM file with an ASSIGN or COMMON command, then using the T parameter on the CATALOG command. Refer to the NOS 2 Reference Set, Volume 3, for more information concerning these commands.

The following list provides brief descriptions of SYSEDIT directives acceptable in LIBDECK. Complete descriptions of all SYSEDIT directives are in the NOS 2 System Maintenance Reference Manual. A list of valid record types follows the directives.

Directive Format	Significance
*AD,nn,$ty_1/rec_1,ty_2/rec_2,\ldots,ty_n/rec_n$	Specifies the alternate device to be used in addition to the system device(s) for storing ABS, OVL, PP, and REL type records. nn is either the EST ordinal or the equipment type, ty_i is the record type, and rec_i is the record name.
*CM,$ty_1/rec_1,ty_2/rec_2,\ldots,ty_n/rec_n$	Defines record rec_i of type ty_i as being central memory resident; legal only for types ABS, OVL, or PP.
*FL,$ty_1/rec_1-fl_1,ty_2/rec_2-fl_2,\ldots,ty_n/rec_n-fl_n$	Record rec_i of type ty_i is loaded with a field length specified by fl_i (fl_i is field length divided by 100_8).
*MS,$ty_1/rec_1,ty_2/rec_2,\ldots,ty_n/rec_n$	Defines record rec_i of type ty_i as being mass storage resident. This is the default residence for routines with no storage area specified in LIBDECK.
*PROC,$rec_1,rec_2,\ldots,rec_n$	Defines record rec_i of type PROC as a procedure file.
*SC,$ty_1/rec_1,ty_2/rec_2,\ldots,ty_n/rec_n$	Defines record rec_i of type ty_i as product set format commands. The command parameters are processed in product set format (refer to the NOS 2 Reference Set, Volume 3).

The following record types may be specified in SYSEDIT directives. Some directives do not allow all types.

Type (ty_i)	Description
ABS	Multiple entry point overlay.
CAP	Fast dynamic load capsule.
OPL	Modify old program library deck.
OPLC	Modify old program library common deck.
OPLD	Modify old program library directory.
OVL	Central processor overlay.
PP	Peripheral processor program.
PPU	Peripheral processor unit program.
PROC	Procedure.
REL	Relocatable central processor program.
TEXT	Unrecognizable as one of the other types.
ULIB	User library program.

BINARY PATCH UTILITIES 11

BINEDIT and BNP are, respectively, the on-line and off-line binary patch utilities, which enable you to patch selected records in CTI or HIVS.† BINEDIT must run under the control of an operating system (on-line). BNP, on the other hand, must run in the absence of an operating system (off-line). After patching records in CTI or HIVS, deadstart from the patched CTI/HIVS tape (refer to Deadstarting from a Patched CTI/HIVS Tape in this section).

ON-LINE BINARY PATCH UTILITY (BINEDIT)

BINEDIT is called by a command in a batch job, by an operator under DIS, or by a command from a terminal. BINEDIT accepts interactive directive input when called from a terminal.

The BINEDIT command has the following format:

```
BINEDIT,P=oldrec,N=newrec,I=dir,L=list,A.
```

Parameters are order-independent.

Parameter	Meaning
P=oldrec	Read old records from file oldrec. The default file name is OLD.
N=newrec	Write new records on file newrec. The default file name is NEW.
I=dir	Get directives from next record on file dir. The default file name is INPUT.
L=list	List output on file list. The default file name is OUTPUT.
A	Abort after encountering any error (fatal or nonfatal). If A is omitted, BINEDIT aborts only when it encounters fatal errors.

BINEDIT DIRECTIVES

BINEDIT directives identify the record and the corresponding words to be patched within the record. The directives can reside in file INPUT or in any local file. BINEDIT directives have special format restrictions:

- A prefix character must appear in character position 1 of all directives, except that the directive following a REPLACE directive has no prefix character. The prefix character must be an asterisk (*), unless explicitly changed by a PREFIX directive.

- The directive name must begin in character position 2. The directive following a REPLACE directive has no directive name.

---

†Customer engineers also use BINEDIT and BNP to patch MSL.

- The directive name and the parameters are separated by characters having 6-bit display code values greater than 54_8; that is, none of the following is a valid separator:

  A through Z 0 through 9 + - * / ( ) $ =

- Numeric parameter fields are decimal unless otherwise indicated.
- Embedded blanks are not permitted within a parameter, but any number of blanks can follow the directive name and any of its parameters.
- Only the directive following the REPLACE directive can span lines.
- Parameters are order-dependent.
- The directive identifier (dirid) must appear in character positions 73 through 80.

Under the following directive header, dirid represents the directive identifier.

Directive		Function
pNAME,rtype/rname,rident,rcksum or pN,rtype/rname,rident,rcksum	dirid dirid	Processes the binary record from the old record file (oldrec) whose record type matches rtype and whose name matches rname. Legal record types are in PP code (12-bit) or in absolute central processor (CP) code (60-bit ABS or OVL). If no record is found, a fatal error occurs. This directive terminates any previous NAME processing.

Parameter	Description
p	Prefix character.
rtype	Record type; must be PP or CP.
rname	Record name.
rident	Record identifier.
rcksum	Record checksum.
dirid	Directive identifier.

Parameters rident and rcksum are optional. If specified, rident is compared with the record patch identifier in the DATE field of the loader prefix (PRFX or 7700) table. rcksum is compared with the calculated checksum of the record read from oldrec. If either set does not match, BINEDIT writes an informative message and continues, unless the A parameter was specified on the BINEDIT command.

Directive		Function
pIDENT,patchid or pI,patchid	dirid dirid	Writes the patch identifier patchid in the DATE field of the loader prefix table of the record written to the new record file (newrec). If no IDENT directive is encountered, a patch identifier of all blanks is written.

Parameter	Description
p	Prefix character.
patchid	Patch identifier.
dirid	Directive identifier.

pCHKSUM,chksum or pC,chksum	dirid dirid	Compares the expected checksum with the calculated checksum of the record written to the new record file (newrec). If they do not match, an informative message appears.

Parameter	Description
p	Prefix character.
chksum	Expected checksum.
dirid	Directive identifier.

p/ comments	dirid	Copies the comments field to the output list file. The comments field cannot exceed 69 characters.

Parameter	Description
p	Prefix character.
comments	User´s comments.
dirid	Directive identifier.

pPREFIX,char or pP,char	dirid dirid	Changes the prefix character in character position 1 to char for subsequent directives. The default is an asterisk (*); a blank is not a valid entry for char.

Parameter	Description
p	Prefix character.
char	New prefix character.
dirid	Directive identifier.

Directive		Function
pREAD,altdir	dirid	Stops BINEDIT from reading directives from file dir and initiates reading them from file altdir at its current position. File altdir is read until end-of-record is encountered, after which BINEDIT resumes reading file dir. A fatal error occurs if this directive appears on file altdir.

Parameter	Description
p	Prefix character.
altdir	Additional directive file.
dirid	Directive identifier.

Directive		Function
pREPLACE,fwa or pR,fwa	dirid dirid	Replaces words beginning at the first word address fwa of the the record to be patched with the words in the next directive. The length of the record is extended if fwa is equal to the last word address plus 1 (lwa + 1). fwa must be greater than the value specified in the preceding REPLACE directive for this record group. By default, fwa is octal. The postradix D specifies that fwa is decimal. If the next directive does not contain a list of words for replacement, a fatal error occurs.

Parameter	Description
p	Prefix character.
fwa	First word address of the record to be patched as it would appear if the word were loaded into memory.
dirid	Directive identifier.

Directive		Function
$word_1$,$word_2$,...,$word_n$.	dirid	Immediately follows a REPLACE directive and provides the new content for the record being patched. $word_i$ is either a 4-digit number (12-bit PP code) or a 20-digit number (60-bit absolute CP code). By default, $word_i$ is octal. The postradix D indicates a decimal entry. A maximum of 10 decimal digits can be specified for CP code. Although no parameter $word_i$ can span lines, the modification directive itself can do so. A period must follow $word_n$.

Directive	Function
	The $word_i$ parameters replace words in the record beginning at address fwa, specified in the REPLACE directive. The length of the record is increased if the modifications extend beyond the last word address plus 1 (lwa + 1).

Parameter	Description
$word_i$	Replacement for a word on the file oldrec.
dirid	Directive identifier.

BINEDIT EXAMPLE

Assume that an error has been found in the first CTI record, IPL, and that the SHN 6 instruction at address 2403 is to be replaced by one SHN 4 instruction. The following job patches the record and writes it to a file, which you can use as input to create a new deadstart tape or a system library file. CURRENT is the permanent file that contains the CTI module.

Job		Comments
job command.		
USER,username,password,familyname.		
CHARGE,*.		
GET,CURRENT.		
BINEDIT,P=CURRENT,N=NEWIPL.		Causes BINEDIT to patch records on CURRENT and to write the modified records to NEWIPL. Patching directives are to be read from the next record.
REWIND,CURRENT,NEWIPL.		
LIBEDIT,P=CURRENT,B=NEWIPL,N=CURRNT2.		Creates a library file containing the patched records and a permanent file containing the updated version of CTI.
SAVE,CURRNT2.		
--eor--		
*IDENT 060580A	CTI01.1	Specifies the patch identifier.
*NAME PP/IPL, 04/16/80 , 1617	CTI01.2	Identifies the record to be patched.
*/ THIS PATCH FIXES PSR CTI01	CTI01.3	Inserts a comment.
*/	CTI01.4	
*REPLACE 2403	CTI01.5	Identifies address to be patched.
1004.	CTI01.6	Identifies the new content of the address specified on REPLACE directive.
*CHKSUM 3612	CTI01.7	Specifies the expected checksum.
--eor--		
*REPLACE IPL		This LIBEDIT directive creates a new library file CURRNT2 with the patched IPL record.
--eoi--		

BINEDIT generates the following list output:

Output	Comment
*IDENT 060580A	CTI01.1
*NAME PP/IPL, 04/16/80 , 1617	CTI01.2
*/ THIS PATCH FIXES PSR CTI01	CTI01.3
*/	CTI01.4
*REPLACE 2403	CTI01.5
1004.	CTI01.6
*CHKSUM 3612	CTI01.7

```
EDITING BEGUN

BEGIN PATCH OF PP/IPL OLD IDENT = 04/16/80
CONTENTS OF ADDRESS 2403 CHANGED
FROM 1006 TO 1004
CHECKSUM CHANGED OLD CHECKSUM=1617 NEW CHECKSUM=3612
END PATCH OF PP/IPL NEW IDENT=060580A
EDITING COMPLETE
```

Informative messages documenting the record and associated addresses patched by BINEDIT.

OFF-LINE BINARY PATCH UTILITY (BNP)

Binary patch utility BNP is restricted to stand-alone, off-line execution. The following paragraphs describe the BNP installing procedure and directives.

INSTALLING PROCEDURE

Use the following procedure to install BNP.

1. Mount the BNP tape, the CTI/HIVS binary tape (input tape), and a scratch tape (output tape) on three tape units. All three tape units must be the same type (66x or 67x) and have the same track type (seven- or nine-track). If only two units are available, mount the BNP tape and the CTI/HIVS binary tape. When the BNP deadstart is complete, remove the BNP tape, and mount the scratch tape on that unit.

2. Deadstart using the BNP tape (refer to the deadstart panel settings for warmstart in the NOS 2 Operator/Analyst Handbook). The following message appears:

```
BNP -   BINARY PATCH UTILITY
        PARAMETERS. PRESS CARRIAGE
        RETURN.

Acc
```

   cc is the BNP revision level.

3. Press carriage return (CR). The following sequence of console displays appears.

```
TAPE TYPE   (1=60X/65X) =  3
            (2=66X,3=67X)

TAPE CHANNEL NO. =  13

EQUIPMENT NO. =  00

INPUT UNIT NO. =  00

OUTPUT UNIT NO. =  01

TAPE MODE (1=7TRK,2=9TRK)  2
```

4. Either press CR to accept the displayed default value or, to change the display default, enter equipment types according to the site configuration.† Press CR after each entry. The equipment number for tape type 66x is 00. For tape type 67x, the equipment number ranges from 00 to 07; the default is 00.

PATCHING DIRECTIVES

After you select the equipment options, the console lists the patching directives with a brief explanation of each (figure 11-1).

```
BNP -  BINARY PATCH UTILITY.  ALL
       ENTRIES FOLLOWED BY (CR)
       EXCEPT (SPACE BAR).

*N,NAME
       COPY RECORDS FROM INPUT TAPE
       TO OUTPUT TAPE UNTIL NAME
       IS FOUND.

*R,WN1,X,X,X,X,X
       REPLACE UP TO 5 12-BIT WORDS
       X. WN1=ADDRESS OF CM WORD OR
       FWA OF PP WORD(S) TO MODIFY

(SPACE BAR)
       COPY MODIFIED RECORD FROM CM
       TO OUTPUT TAPE.

G -    COPY INPUT TAPE TO OUTPUT
       TAPE TO (EOI).

R -    REWIND TAPE UNITS.
```

Figure 11-1. BNP Patching Directives Display

†NOS does not support 60x and 65x tape units.

A more detailed explanation of the patching directives follows. These directives allow you to patch the CTI/HIVS binary tape to create a new CTI/HIVS binary tape. Terminate each patching directive with a carriage return (CR).

Name Directive (*N)

The name directive copies records from the CTI/HIVS tape to the scratch tape. It copies from the current position of CTI/HIVS to the requested record name. The name directive then loads the requested record into CM so modifications can be made.

Directive	Function
*N,name	Specifies the name of the record to be modified; name is from one through seven alphanumeric characters. If more than one record is to be modified, you must modify them in the order they appear on the tape, because the name directive copies all records preceding the record name to the output tape.

Press CR; one of the following messages appears.

Message	Significance
LOADED RECORD name	Requested record is loaded into CM and is ready for modification.
name NOT FOUND	Either requested record name is not on the tape, or the tape is positioned beyond the record specified. Verify the record name. If it is correct, rewind the tapes (R) and restart the patching of the CTI/HIVS tape.
OSB FOUND - COPY HALTED	Copying halted because the operating system bootstrap (OSB) record was encountered. A CTI/HIVS tape does not have an OSB record. Make sure the CTI/HIVS tape is on the input tape unit.

After the system has loaded the requested record to CM, enter the replace directive.

Replace Directive (*R)

The replace directive replaces CP and PP words on the CTI/HIVS tape. When the record to be patched is loaded to CM, enter the following directive.

Directive	Function
*R,wn1,x_1,x_2,...,x_5	Replaces from one through five 12-bit octal words starting at address wn1. x_i are 12-bit octal words. When modifying a CP program, you must enter all five 12-bit octal words; if you enter less than five, zero entries are recorded. When modifying a PP program, you can enter from one through five 12-bit octal words.

For each patch of the record, enter a replace directive. When patching of the record is complete, enter the write directive to write the patched record to the scratch tape.

Press CR; the following message appears.

RECORD MODIFIED

Write Directive (Space Bar)

The write directive writes the modified record from CM to the scratch tape. After modifying a record, press the space bar. The following message appears.

```
RECORD COPIED CHECKSUM = xxxx
```

xxxx is the new checksum of the modified record.

After the patched record is written to the scratch tape, request a new record with the name directive or, if patching of the file is complete, enter the copy directive.

Copy Directive (G)

The copy directive copies all remaining records on the CTI/HIVS tape to the output tape. When patching is complete and the last record modified is not the last record of the tape, enter the copy directive:

```
G
```

Press CR; one of the following messages appears.

Message	Significance
COPY COMPLETED	All records copied to the output tape.
OSB FOUND - COPY HALTED	Copying halted because an OSB record was encountered. A CTI/HIVS tape does not have an OSB record. Make sure the CTI/HIVS tape is on the input tape unit.

Rewind Directive (R)

If the CTI/HIVS or scratch tape is mispositioned, enter the rewind directive:

```
R
```

The rewind directive rewinds both tapes.

DEADSTARTING FROM A PATCHED CTI/HIVS TAPE

Use the patched CTI/HIVS tape to perform a deadstart for an operating system located on either disk or tape.

CTI/HIVS DEADSTART FROM DISK

The operating system file must be on disk. If CTI is patched, install it on disk (refer to Step 4 - Install CTI Module in Section 2); if HIVS is patched, install it on disk (refer to Step 6 - Install HIVS Module on Disk in section 2). Deadstart from that disk (refer to the NOS 2 Operator/Analyst Handbook).

CTI/HIVS DEADSTART FROM TAPE

To deadstart from the patched CTI/HIVS tape use the following procedure.†

1. Mount the patched CTI/HIVS tape without the write enable ring and ready the unit.

2. Set the deadstart panel for warmstart from tape (refer to the NOS 2 Operator/Analyst Handbook).

3. Activate the deadstart switch. The CTI initial options (A) display appears (refer to figure 2-1).

4. Select either the (CR) or 0 option (refer to figure 2-1). Refer to the NOS 2 Operator/Analyst Handbook for a description of deadstart with operator intervention (0 option). When CTI deadstart is complete, the following message appears.

   ```
   OSB NOT FOUND ON DEVICE
   ENTER ALTERNATE DEVICE TYPE - m
   (1=66x,2=67x,3=DISK)
   ```

5. If CTI is on disk, enter 3. If CTI is on tape, enter 1 or 2.

6. Press CR. The system now requests the channel, equipment, and unit numbers for the disk that contains the operating system deadstart file. Enter the channel, equipment, and unit numbers for the device and press CR after each entry.

7. When deadstart is complete, the initial operating system display appears.

---

†For models 825, 835, or 855, HIVS must be installed on disk in order to deadstart NOS.

MESSAGES A

This appendix contains an alphabetical listing of the messages that may appear at the system console or a terminal when using BINEDIT or BNP. Messages beginning with variable names or characters are listed alphabetically according to the first nonvariable word or character. For example, the message

    name NOT FOUND

is alphabetized starting with the nonvariable NOT.

SEE ERRATA

MESSAGE	SIGNIFICANCE	ACTION	ROUTINE
ADDRESS address OUT OF RANGE.	The REPLACE directive specifies an address that is either less than the load address of the record or greater than the load address plus program length (that is, the last word address plus one).	Ensure that all addresses in the REPLACE directives are within their proper ranges and retry.	BINEDIT
COPY COMPLETED	All records have been copied to the output tape.	None.	BNP
COPY ERR name	System could not complete the requested copy of name (a program or command buffer).	Press space bar and copying continues with the next program or command buffer. The program or command buffer the error occured in will have to be recopied.	CTI
DIRECTIVE TRUNCATED	An input directive exceeds 80 characters. The system truncates the directive to 80 characters. BINEDIT aborts only if the A parameter appears in the BINEDIT command.	If the job aborted, ensure that input directives do not exceed 80 characters and rerun the job. If the job did not abort and data was lost, shorten the record; otherwise, no action is required.	BINEDIT
DIRECTORY FULL	The CDA directory is full; it can only have 64 entries.	Reinstall the CDA utility and reload data and programs to the CDA, without exceeding the limit of 64 CDA directory entries.	CTI
DISK BUSY.	No read or write was performed because the disk was busy.	Press the space bar to retry disk access.	CTI
DISK CPLR RSVD.	System cannot access the disk controller because the disk is shared in a multimainframe environment and another mainframe is using it. The system continues its attempts to access the disk controller. This message will be repeated until the disk is successfully accessed.	None.	CTI
DISK FUNCTION REJ name function.	Disk rejected the attempted function. name is the current program or command buffer name.	Press the space bar to retry the function.	CTI
DISK FUNCTION REJECT. FUNCTION = xx	The disk controller did not respond normally to function code xx.	Press CR to retry. If the condition persists, contact a customer engineer.	CTI
DISK RESERVED	The disk unit is reserved by another controller. Automatic retry is initiated when the reserved status of the disk unit is cleared.	Clear the reserve status of the disk unit. If reserved status persists, contact site analyst or customer engineer.	CTI
DISK STATUS ERROR STATUS = xxxx	The disk controller indicates an error condition. The status word xxxx indicates the type of error.	Press CR to retry; if the condition persists, contact site analyst or customer engineer.	CTI
DISK UNIT RSVD.	The controller cannot access the disk because another controller is using it. The controller continues its attempts to access the disk. This message will be repeated until the disk is successfully accessed.	None.	CTI
DUPLICATE LFN lfn	The file is already in use.	Specify another file and retry.	BINEDIT
DUPLICATE PARAMETER xxxx	Parameter xxxx appears more than once in the BINEDIT command.	Discard one occurrence of the duplicate parameter and retry.	BINEDIT
EMPTY FILE filename.	The system encountered the end of the file before it read any data from input file filename.	Ensure that the input file is not empty and then retry.	BINEDIT
EMPTY MACRO CALL FILE	SEE ERRATA		
ERROR IN MACRO CALL FILE	"		
ERROR IN MACRO TEXT FILE	"		
ERROR IN USEROPS FILE	"		

MESSAGE	SIGNIFICANCE	ACTION	ROUTINE
FLAW CYL cyl TRK trk SEC sec.	In an attempt to copy from tape to disk, a flaw was found on the disk at cylinder number cyl, track number trk, and sector number sec. The requested copy was not completed.	Press the space bar to continue the copy request. The flawed sector will be skipped.	CTI
FORMAT ERROR	Incorrect keyboard entry.	Reenter the directive.	BNP
ILLEGAL ENTRY.	The entry for disk or tape channel, tape equipment, or tape unit is not an acceptable value.	Press the space bar. After the parameter options and prompt are displayed, reenter a value for the parameter.	CTI
INCOMPLETE SYMBOL	SEE ERRATA		
INVALID DIRECTIVE	A directive occurs out of sequence or the directive name is in error.	Ensure that the directives are in the proper sequence and that the directive names are correct. Then retry.	BINEDIT
INVALID PARAMETER xxxx	Parameter xxxx in the BINEDIT command has an incorrect format or an incorrect name.	Ensure that the parameters have correct names and proper formats and retry.	BINEDIT
LINE TOO LONG AFTER SUBSTITUTION	SEE ERRATA		
LOADED RECORD name	Requested record is loaded into CM and ready for BNP modification.	Enter the REPLACE directive.	BNP
MODIFICATION DIRECTIVE EXPECTED	A modification directive did not follow a REPLACE directive or the preceding modification directive did not terminate with a period.	Ensure that all modification directives terminate with a period and that a modification directive follows each REPLACE directive. Then retry.	BINEDIT
NEW CHECKSUM MISMATCH	The expected checksum specified on the CHKSUM directive does not match the calculated checksum of the modified record written to the new record file. Refer to the contents of NEW CHECKSUM in the preceding CHECKSUM CHANGED line of the output report. BINEDIT aborts only if the A parameter appears in the BINEDIT command.	If the job aborted, ensure that the two checksums match and then rerun the job; otherwise; no action is required.	BINEDIT
NULL SYMBOL	SEE ERRATA		
NO NAME DIRECTIVE	A NAME directive must appear before the CHKSUM, REPLACE, and modification directives.	Ensure that a NAME directive precedes the other directives and retry.	BINEDIT
NO REPLACE DIRECTIVE	The user must specify a REPLACE directive.	Specify a REPLACE directive and retry.	BINEDIT
NO TERMINATOR	The last word of each modification directive must terminate with a nonblank separator in or before column 72. If this is the last modification line, the terminator must be a period. If it is a continuation line, the terminator must be a nonblank separator other than a period.	Ensure that all modification directives have proper terminators and retry.	BINEDIT
NO WRITE ENABLE	Write ring missing from the tape on the output unit.	Put the write ring on the tape.	BNP
name NOT FOUND	Requested record name is not on the tape or the tape is positioned beyond the record specified.	Verify the record name. If it is correct, rewind the tapes (R directive) and restart the patching of the CTI/HIVS tape.	BNP
OLD CHECKSUM MISMATCH	The expected checksum specified on the NAME directive does not match the calculated checksum of the unmodified record read from the old record file. Refer to the contents of OLD CHECKSUM in the preceding CHECKSUM CHANGED line of the output report. BINEDIT aborts only if the A parameter appears in the BINEDIT command.	If the job aborted, ensure that the two checksums match and rerun the job; otherwise, no action is required.	BINEDIT
OLD IDENT MISMATCH	The expected patch identifier specified on the NAME directive does not match the actual patch identifier contained in the prefix table of the record read from the old record file. Refer to the contents of OLD IDENT in the preceding BEGIN PATCH line of the output report. BINEDIT aborts only if the A parameter appears in the BINEDIT command.	If the job aborted, ensure that the patch identifiers in question match and rerun the job; otherwise, no action is required.	BINEDIT

MESSAGE	SIGNIFICANCE	ACTION	ROUTINE
OSB FOUND - COPY HALTED	Copying halted because the operating system bootstrap (OSB) record was encountered.	A CTI/HIVS tape does not have an OSB record. Make sure the CTI/HIVS tape is on the input tape unit.	BNP
PARAMETER TRUNCATED xxxx	An optional parameter contains too many characters. xxxx is the parameter in truncated form. BINEDIT aborts only if the A parameter appears in the BINEDIT command.	If the job aborted, ensure that the parameter in question has a proper character length and rerun the job; otherwise, no action is required.	BINEDIT
PROGRAM NOT ON TAPE - xxxx	The program binary, whose 4-character binary mnemonic is xxxx, is not on the tape that is being read.	Verify program mnemonic and check to see that the correct tape is mounted.	CTI
READ ERROR.	During the read from the input tape, a parity error occurred.	Rewind tape and restart the patching of the CTI/HIVS tape.	BNP
RECORD COPIED CHECKSUM=xxxx.	The system copied the record from CM to the scratch tape. xxxx is the new checksum of the modified record.	None.	BNP
RECORD MODIFIED	The system modified the record.	None.	BNP
RECORD NOT FOUND rtype/rname	The system could not find on the old record file a record matching the name (rname) and type (rtype) defined in the NAME directive.	Ensure that the name and type of records defined in the NAME directive match those of the records in the old record file and retry.	BINEDIT
SFM CURRENT ATTRIBUTE IS NOT IN LID TABLE.	A LID entry was requested to be altered, but the table differs from the entry being used by the caller.	Check the LID table and retry.	SFM
SFM - LID NOT LEGAL.	An attempt was made to alter a LID that does not exist in the LIDT.	None.	SFM
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.	Make the buffer larger and rerun.	SFM
SFM - LID TABLE TOO LONG FOR BUFFER.	The installer has requested the LID table with a buffer which is too small.	Specify minimum buffer size (MXLID in NOSTEXT).	SFM
SYMBOL TOO BIG [illegible]			
TAPE ERR STAT status.	Tape drive error of type status occured during the attempted copy. Printed on the preceding line is the name of the current program or command buffer name.	Press space bar to continue the copy. There may be errors in the copy and another copy may have to be made.	CTI
TAPE FUNC REJ function.	Tape drive rejected the attempted function. Printed on the preceding line is the name of the current program or command buffer.	Press the space bar to retry the function.	CTI
TAPE STATUS ERROR	Tape unit not ready or load failure.	Check tape unit.	BNP
TAPE STATUS ERROR STATUS = xxxx	The tape controller indicates an error condition. The status word xxxx indicates the type of error.	Press CR to retry. If the condition persists, contact site analyst or customer engineer.	CTI
TOO MANY LID ENTRIES.	During deadstart, the LIDT became full while entering the LIDS specified in IPRDECK.	Redeadstart and specify fewer LIDs or specify larger LIDT in CMRDECK.	SFM
UNIT xx NOT RDY - OP ABORT	Tape unit xx is not ready.	Ready the tape unit. The input and output tapes are automatically rewound. Restart the patching of CTI/HIVS tape.	BNP
UNUSABLE DISK	The cylinder on which HIVS was to be copied has a flaw.	Redeadstart and install CTI and HIVS to a different device.	CTI
WRITE ERROR	During the write to the output tape a parity error occurred.	Rewind the tapes and restart the patching of the CTI/HIVS tape.	BNP

GLOSSARY

B

AAM	Advanced Access Methods.
ABC	Automatic buffer controlware loader.
ABL	Application block limit.
ACPD	Analyze collected performance data.
ADL	Application Definition Language.
AIP	Application Interface Program.
ALGOL	Algorithmic Language.
ANSI	American National Standards Institute.
APL	A Programming Language.
APRDECK	Auxiliary mass storage parameter deck.
ARF	After-image recovery file.
ASCII	American Standard Code for Information Interchange.
BAM	Basic Access Methods.
BASIC	Beginner's All-Purpose Symbolic Instruction Code.
BINEDIT	Binary Editor.
BIO	Central site batch I/O.
BNP	Binary Patch.
BR	Backup requirement.
BSC	Binary synchronous communications.
CAS	Conversion Aids System.
CCG	Common Code Generator.
CCL	CYBER Control Language.
CCP	Communications Control Program.
CDA	CTI/MSL disk area.
CDCS	CYBER Database Control System.
CEDIAG	Customer Engineer Diagnostics.
CID	CYBER Interactive Debug.
CLA	Communications line adapter.
CM	Central memory.
CMM	Common Memory Manager.
CMR	Central memory resident.
CMRDECK	Central memory resident deck.
CMS	Check mass storage.
CMU	Compare and move unit.
COBOL	Common Business-Oriented Language.
COMPASS	Comprehensive Assembler.
CPCOM	Central program communication.
CPD	Collect performance data.
cpi	Characters per inch.
cps	Characters per second.
CPU	Central processing unit.
CRF	Communication recovery file.
CRM	CYBER Record Manager.
Cross	CYBER Cross System.
CS	Communications Supervisor.
CSU	Cartridge storage unit.
CTI	Common testing and initialization.

DAT Device access table.

DBU Database Utilities.

DDL Data Description Language.

DDP Distributive data path.

DIS Job display routines.

DIT Device information table.

DLFP Debug log file processor.

DPB Default parameter binaries.

DSD Dynamic system display.

EBCDIC Extended Binary-Coded Decimal Interchange Code.

ECO Engineering change order.

ECS Extended core storage.

EI Environment interface.

EJT Executing job table.

EOI End of information.

ESM Extended semiconductor memory.

EST Equipment status table.

FCL FORTRAN Common Library.

FCO Field change order.

FCP File conversion processor.

FCS File Conversion Aids System.

FDBF FORTRAN Data Base Facility.

FDP Pack formatting driver.

FET File environment table.

FLPP First-level peripheral processor.

FNT File name table.

FORM File Organizer and Record Manager.

FORTRAN Formula Translation.

FOT Family ordinal table.

FST File status table.

GCR Group-coded recording.

GE Group encoded.

HIP Host Interface Program.

HIVS Hardware Initialization and Verification Software.

IAF Interactive Facility.

ICPD Initiate collect performance data.

ID Identification.

IMF Information Management Facility.

I/O Input/output.

K Represents 1024.

LCF Local configuration file.

LCME Large central memory extended.

LCN Loosely coupled network.

LCP Language conversion processors.

LCS Language Conversion Aids System.

LFG Load file generator.

LID Logical identifier.

LIP Local Interface Program.

MAG Magnetic tape subsystem.

MCS Message Control System.

MMF Multimainframe.

MRT Machine recovery table.

MSA Mass storage adapter.

MSC Mass storage coupler.

MSF	Mass storage facility.
MSL	Maintenance Software Library.
MSS	Mass Storage Subsystem.
MST	Mass storage table.
MST	Mass storage test.
MUX	Multiplexer.
NAD	Network Access Device.
NAM	Network Access Method.
NAMI	Network access method initialization program.
NBL	Network block limit.
NCF	Network configuration file.
NCT	Network communication table.
NDA	NPU dump analyzer.
NDL	Network Definition Language.
NDLP	Network Definition Language Processor.
NETUVSN	Update VSN program.
NIP	Network interface program.
NOS	Network Operating System.
NPU	Network processing unit.
NS	Network Supervisor.
NVF	Network Validation Facility.
OSB	Operating system bootstrap.
OVCAP	Overlay capsule.
PAD	Packet assembly/disassembly.
PF	Preserved file.
PFM	Permanent file manager.
PIP	Peripheral interface program.
PL	Program library.
PMD	Postmortem dump.
PP	Peripheral processor.
PPCOM	PP systems communications.
PR	Preferred residence.
PRU	Physical record unit.
PVT	Pattern value table.
QFT	Queued file table.
RBF	Remote Batch Facility.
RDF	Remote Diagnostic Facility.
RHF	Remote Host Facility.
S/C	Storage control.
SIP	Stimulator interface program.
Sort	Sort/Merge.
SRU	System resource unit.
SYMPL	Symbolic Programming Language.
TAF	Transaction Facility.
TIP	Terminal Interface Program.
TRT	Track reservation table.
TVF	Terminal Verification Facility.
UEM	Unified extended memory.
USASI	United States of America Standards Institute; now known as ANSI.
USLIB	User library.
VSN	Volume serial number.
XEDIT	Extended Text Editor.

RELEASE MATERIALS DESCRIPTION

C

Sites can order the tapes released with the base operating system package and the optional products as:

- Seven-track, 800 cpi.
- Nine-track, 800 cpi.
- Nine-track, 1600 cpi.

The format of all released tapes, except the HIVS and BNP tapes, is:

- 64-character set mode.
- Internal format (F=I).
- Labeled with the file identifier, as shown in table C-1, in the HDR1 label.

The format of the HIVS and BNP tapes is:

- Binary mode.
- System internal format (F=SI).
- Unlabeled.

Table C-1. Release Tapes File Identifiers (Sheet 1 of 2)

VSN	File ID in HDR1†	VSN	File ID in HDR1†
REL0	INSTALL*NOSlevel	REL2B	CED1P2*NOSlevel
REL1A	OPL2P1*NOSlevel	REL3A	CPS3P6*NOSlevel
REL1D	XED3P1*NOSlevel	REL3F	CID1P1*NOSlevel
REL1F	MMF1P0*NOSlevel	REL4A	FTN4P8*NOSlevel
REL1G	TRC1P0*NOSlevel	REL4B	FTNI4P8*NOSlevel
REL1J	HSIO1P0*NOSlevel	REL4C	~~FCL4P7~~*NOSlevel FCL4P8
REL2A	TOOLS*NOSlevel		

†Replace level in each file identifier with 580577 (for example, the identifier for release tape REL1A is OPL2P1*NOS580577).

Table C-1. Release Tapes File Identifiers (Sheet 2 of 2)

VSN	File ID in HDR1†	VSN	File ID in HDR1†
REL4D	FDBF1P3*NOSlevel	REL11H	DDL3P2*NOSlevel
REL4E	FTN5P1*NOSlevel	REL12C	TAF1P2*NOSlevel
REL4F	F451P0*NOSlevel	REL12E	IAF1P0*NOSlevel
REL4G	FCL5P1*NOSlevel	REL12F	MCS1P0*NOSlevel
REL5B	CAS3P0*NOSlevel	REL12G	NAM1P5*NOSlevel
REL5C	COB5P3*NOSlevel	REL12H	RBF1P5*NOSlevel
REL6A	SORT4P6*NOSlevel	REL13A	CCS1P2*NOSlevel
REL6B	SORT5P0*NOSlevel	REL13B	CCP3P5*NOSlevel
REL7B	ALG5P1*NOSlevel	REL13C	CCP2550*NOSlevel
REL8A	BAS3P5*NOSlevel	REL13E	OLD1P0*NOSlevel
REL8B	APL2P1*NOSlevel	REL13F	RNPU1P0*NOSlevel
REL8C	PAS1P0*NOSlevel	REL14A	PLI1P0*NOSlevel
REL11A	DC21P0*NOSlevel	REL14B	CCG1P0*NOSlevel
REL11B	IMF1P1*NOSlevel	REL14C	MSS1P0*NOSlevel
REL11D	DBU1P2*NOSlevel	REL16A	RHF~~2P0~~*NOSlevel
REL11E	QU3P4*NOSlevel	REL16B	RHP~~1P0~~*NOSlevel
REL11G	CDCS2P3*NOSlevel		

†Replace level in each file identifier with 580577 (for example, the identifier for release tape REL1A is OPL2P1*NOS580577).

The base operating system is released on the following tapes.

Tape Label	Tape Contents
BNP	Binary for the off-line binary patch utility (BNP). Only install BNP if you receive a binary patch order.
HIVS	Binary for the hardware installation verification software (HIVS).
Deadstart tape	Binaries for NOS.
RELO	Installation decks program library, PSR reports, suggested code, and NOS corrective code.

Tape Label	Tape Contents
REL1A	Program library for NOS.
REL2B	Binary and program library for CEDIAG.
REL3A	Binaries and program libraries for COMPASS, Update, CYBER Utilities, Common Memory Manager, 819 PPU driver, CYBER Control Language, CYBER Record Manager Basic Access Methods, BINEDIT, 8-Bit Subroutines, FORM, CYBER Record Manager Advanced Access Methods, Product Texts, and CYBER Loader.

NOTE

The modification sets and resequenced decks for the operating system are sent out as a separate tape for a system release if they cannot be written on the REL1A tape.

All other tapes listed in table C-1 represent packages that must be ordered separately from the NOS package. The maintenance package, including the following products, is one such package.

Tape Label	Tape Contents
REL2A	Binary and program libraries in Modify format for Maintenance Tools, and binaries and program libraries in Update format for 881/883 Pack Formatting and SYMPL.

The rest of this section contains descriptions of the tapes. Each description includes a listing of the files on each tape, and states if the program library is in Modify or Update format. The descriptions are divided into the following three groups:

- Base operating system.
- Maintenance package.
- Optional products.

Within each group the tapes are listed by their tape labels in numerical order.

Most of the tapes have either four files or a multiple of four files. The first file contains the product's program library, and the second file contains the product's absolute binary code. The third file varies but often contains the product's relocatable binary code or is empty. The fourth file is the product maintenance file, which contains the relocatable binaries that generated the ABS and OVL type records for that product. For further information concerning the contents of this file, consult a listing of DECKOPL for each product (refer to Step 8 - Set Up Installation Files in section 2). In addition, the following is also true.

- For Update-formatted products (except Cross, CCP, and Conversion Aids System), file USER is also saved on the product maintenance file.
- For Modify-formatted products, the modification sets used to generate the program library for that product, as well as the resequenced decks, are contained on the product maintenance file.

BASE OPERATING SYSTEM

Tape Label	Tape Contents
BNP	BNP contains the binaries of the off-line binary patch utility (BNP). Only install BNP if you receive a binary patch order.
Deadstart tape	The released deadstart tape contains binaries of:

- NOS
  - Modify
  - Text Editor
- COMPASS
- SYMPL
- CYBER Control Language
- CYBER Loader
- CYBER Record Manager
  - Basic Access Methods
  - Advanced Access Methods
- FORM
- Product Texts
- Product Texts I/O
- Update
- CYBER Common Utilities
- Common Memory Manager
- Controlware
- Maintenance Tools
- 8-Bit Subroutines
- 881/883 Pack Formatting
- CTI
- BINEDIT

The deadstart tape is unlabeled, either seven-track (800 cpi) or nine-track (1600 cpi), in binary recording mode, and one file.

Tape Label	Tape Contents
REL0	REL0 is used in the installation and modification of the operating system and optional products. It has six files. During installation these files are copied to permanent files (refer to step 1 of Step 8 - Set Up Installation Files in section 2).
	File 1 — Procedure to install the REL0 files, binary of REP, and installation decks program library in Modify format. The permanent file name is DECKOPL.
	File 2 — Operating system corrective code, if any; text record with Modify directives. The permanent file name is MDYMODS.
	File 3 — Optional products modifications, if any, in Update format. The permanent file name is CPRD.
	File 4 — PSR data base, if any.
	File 5 — Miscellaneous code, if any, in Update format. The permanent file name is MISCPL.
	File 6 — Network Host Products modifications, if any, in Update format. The permanent file name is CNSP.
REL1A	REL1A contains the NOS system old program library in Modify format. It has four files.
	File 1 — Program library for NOS, Modify, and Text Editor in Modify format.
	File 2 — Empty file.
	File 3 — Empty file.
	File 4 — Product maintenance file.
REL3A	REL3A contains program libraries in Update format for the standard product set in the base operating system. It has 48 files.
	File 1 — Program library for COMPASS.
	File 2 — Absolute binary code of COMPASS.
	File 3 — Binaries that are placed on the library file SYSLIB.
	File 4 — Product maintenance file.
	File 5 — Program library for Update, CYBER Common Utilities, and Common Memory Manager.

Tape Label		Tape Contents
	File 6	Absolute binary code of Update and CYBER Common Utilities.
	File 7	Binaries that are placed on library files SYSLIB and SYMLIB.
	File 8	Product maintenance file.
	File 9	Program library for 819 PPU driver.
	File 10	Absolute binary code of 819 PPU driver.
	File 11	Empty.
	File 12	Product maintenance file.
	File 13	Program library for CPCTEXT, IPTEXT, SPPTEXT, SCPTEXT, and CPUTEXT decks.
	File 14	Binaries of CPCTEXT, IPTEXT, SPPTEXT, SCPTEXT, and CPUTEXT.
	File 15	Empty file.
	File 16	Product maintenance file.
	File 17	Program library for PFMTEXT, CPC, IORANDM, IO, CHEKPT, and RECOVR decks.
	File 18	Binary of PFMTEXT.
	File 19	Binaries of CHEKPT, CPC, RECOVR, IORANDM, and IO that are placed on the library file SYSLIB.
	File 20	Product maintenance file.
	File 21	Program library for CYBER Loader.
	File 22	Binary of CYBER Loader.
	File 23	Binaries of PILOAD, FDL.RES, FDL.MMI, FDL.OCR, FOL.RES, UCLOAD, and TRAPPER.
	File 24	Product maintenance file.
	File 25	Program library for CCL.
	File 26	Absolute binary code of CCL.
	File 27	Empty file.
	File 28	Product maintenance file.
	File 29	Program library for BAM.
	File 30	TXTCRM, IOTEXT, CRMEP, and FILE binaries.

Tape Label		Tape Contents
	File 31	Relocatable binary modules to reside in SYSLIB and relocatable binary modules and capsules to reside in BAMLIB.
	File 32	Product maintenance file.
	File 33	Program library for BINEDIT.
	File 34	Absolute binary code of BINEDIT.
	File 35	Empty file.
	File 36	Product maintenance file.
	File 37	Program library for 8-Bit Subroutines.
	File 38	Absolute binary code of COPY8P.
	File 39	I/O modules binary code for 8-Bit Subroutines.
	File 40	Product maintenance file.
	File 41	Program library for FORM.
	File 42	Absolute binary code of FORM.
	File 43	Library routines for FORM.
	File 44	Product maintenance file.
	File 45	Program library for AAM.
	File 46	Binary code of absolute utilities for AAM.
	File 47	I/O modules binary code for AAM.
	File 48	Product maintenance file.

MAINTENANCE PACKAGE

The maintenance package is contained on the following tape.

MAINTENANCE TOOLS, 881/883 PACK FORMATTING, AND SYMPL1

Tape Label	Tape Contents
REL2A	REL2A contains three program libraries. It has 12 files.

File 1 — Program library in Modify format for:

- Interactive stimulator (STIMULA, 1TS, and DEMUX).
- Dayfile sort program (DFSORT).
- P register analyzer (PSAMP and SMP).
- All games.
- Status/control (S/C)† register maintenance program.
- CPU debugging routine.

File 2 — Binaries of file 1.

File 3 — Empty file.

File 4 — Product maintenance file.

File 5 — Program library in Update format for 881/883 pack formatting.

File 6 — Absolute binary code of FORMAT and pack formatting driver (FDP).

File 7 — Empty file.

File 8 — Product maintenance file.

File 9 — Program library in Update format for SYMPL.

File 10 — Absolute binary code of SYMPL compiler overlays.

File 11 — Relocatable binary code of SYMPL object library.

File 12 — Product maintenance file.

†For models 865 and 875, S/C registers are maintenance registers (MR).

OPTIONAL PRODUCTS

The optional products are contained on the following tapes.

XEDIT 3

Tape Label	Tape Contents
REL1D	REL1D contains a program library in Modify format. It has four files.
	File 1 Program library for XEDIT.
	File 2 Absolute binary of XEDIT.
	File 3 Empty file.
	File 4 Product maintenance file.

MULTIMAINFRAME MODULE 1

Tape Label	Tape Contents
REL1F	REL1F contains a program library in Modify format. It has four files.
	File 1 Program library for Multimainframe Module.
	File 2 Multimainframe Module binaries.
	File 3 Empty file.
	File 4 Product maintenance file.

TRACER 1

Tape Label	Tape Contents
REL1G	REL1G contains a program library in Modify format. It has four files.
	File 1 Program library for Tracer.
	File 2 Tracer binaries.
	File 3 Relocatable binary code of Probe, collect performance data (CPD), analyze collected performance data (ACPD), and initiate collect performance data (ICPD).
	File 4 Product maintenance file.

HIGH SPEED I/O OPTION

Tape Label	Tape Contents	
REL1J	REL1J contains a program library in Modify format. It has four files.	
	File 1	Program library for HSIO.
	File 2	Absolute binary code of HSIO.
	File 3	Empty file.
	File 4	Product maintenance file.

CYBER INTERACTIVE DEBUG 1

Tape Label	Tape Contents	
REL3F	REL3F contains a program library in Update format. It has four files.	
	File 1	Program library for CID.
	File 2	Absolute binary code and overlay records for CID.
	File 3	Relocatable binary of CID.
	File 4	Product maintenance file.

FORTRAN EXTENDED 4

Tape Label	Tape Contents	
REL4A	REL4A contains a program library in Update format. It has four files.	
	File 1	Program library for FORTRAN Extended.
	File 2	Absolute binary code and compiler overlays.
	File 3	Empty file.
	File 4	Product maintenance file.

FORTRAN EXTENDED 4 WITH INTERACTIVE OPTION

Tape Label	Tape Contents	
REL4B	REL4B contains a program library in Update format. It has four files.	
	File 1	Program library for FORTRAN Extended with Interactive Option.
	File 2	Absolute binary code and compiler overlays.
	File 3	Empty file.
	File 4	Product maintenance file.

FORTRAN COMMON LIBRARY 4 WITH POSTMORTEM DUMP UTILITY

Tape Label	Tape Contents
REL4C	REL4C contains two program libraries in Update format. It has eight files.

File 1	Program library for FCL 4 mathematical and I/O routines.
File 2	FCLTEXT and MATHTXT overlay records.
File 3	Relocatable binaries of FCL 4 routines.
File 4	Product maintenance file.
File 5	Program library for PMD routines.
File 6	PMD postprocessor overlay record.
File 7	Empty file.
File 8	Product maintenance file.

FORTRAN DATA BASE FACILITY 1

Tape Label	Tape Contents
REL4D	REL4D contains a program library in Update format. It has four files.

File 1	Program library for FDBF.
File 2	Absolute binary code and overlay records for FDBF.
File 3	Binary code of library routines for FDBF.
File 4	Product maintenance file.

FORTRAN 5

Tape Label	Tape Contents
REL4E	REL4E contains a program library in Update format. It has four files.

File 1	Program library for FORTRAN 5.
File 2	Absolute binary code and compiler overlays.
File 3	Empty file.
File 4	Product maintenance file.

FORTRAN 4 TO 5 CONVERSION AID 1

Tape Label	Tape Contents
REL4F	REL4F contains a program library in Update format. It has four files. File 1 Program library for FORTRAN 4 to 5 Conversion Aid. File 2 Absolute binary code and overlay records for conversion aid. File 3 Empty file. File 4 Product maintenance file.

FORTRAN COMMON LIBRARY 5 WITH POSTMORTEM DUMP UTILITY

Tape Label	Tape Contents
REL4G	REL4G contains two program libraries in Update format. It has eight files. File 1 Program library for FCL 5 mathematical and I/O routines. File 2 FC5TEXT and MTH5TXT overlay records. File 3 Relocatable binaries of FCL 5 routines. File 4 Product maintenance file. File 5 Program library for PMD routines. File 6 PMD postprocessor overlay record. File 7 Empty file. File 8 Product maintenance file.

CONVERSION AIDS SYSTEM 3

Tape Label	Tape Contents
REL5B	REL5B contains two program libraries in Update format. It has 16 files.

File 1	Program library for LCS.
File 2	Absolute load module for LCS.
File 3	Binary (FORTRAN) syntax file for LCS.
File 4	Absolute load module COUP for COSY-to-Update file conversion.
File 5	Absolute load module COPYCOB for COBOL-COPY-library file conversion.
File 6	Binary data file for COSY-to-Update file conversion (file 1 of 2).
File 7	Binary data file for COSY-to-Update file conversion (file 2 of 2).
File 8	Program library for FCS.
File 9	Binary data file for FORTRAN file conversion processor (FCP) verification.
File 10	Binary data file for COBOL FCP verification.
File 11	Absolute load module CBLFCP1 for COBOL FCP.
File 12	Absolute load module CBLFCP2 for COBOL FCP.
File 13	Absolute load module FTNFCP1 for FORTRAN FCP.
File 14	Absolute load module FTNFCP2 for FORTRAN FCP.
File 15	Binary (FORTRAN) syntax file CBLFCPM for COBOL FCP.
File 16	Binary (FORTRAN) syntax file FTNFCPM for FORTRAN FCP.

COBOL 5

Tape Label	Tape Contents
REL5C	REL5C contains a program library in Update format. It has four files.

File 1	Program library for COBOL 5.
File 2	Absolute binary code and compiler overlays.
File 3	Relocatable binary code of library routines.
File 4	Product maintenance file.

SORT/MERGE 4

Tape Label	Tape Contents
REL6A	REL6A contains a program library in Update format. It has four files.
	File 1 — Program library for Sort/Merge 4.
	File 2 — Binaries of SORTMRG and SMTEXT.
	File 3 — Relocatable binary code of library routines.
	File 4 — Product maintenance file.

SORT/MERGE 5

Tape Label	Tape Contents
REL6B	REL6B contains a program library in Update format. It has four files.
	File 1 — Program library for Sort/Merge 5.
	File 2 — Overlays of Sort/Merge 5 (0,0).
	File 3 — Binaries of SRT5LIB.
	File 4 — Product maintenance file.

ALGOL-60 5

Tape Label	Tape Contents
REL7B	REL7B contains a program library in Update format. It has four files.
	File 1 — Program library for ALGOL-60 5.
	File 2 — Absolute binary code and compiler and symbol table overlays.
	File 3 — Relocatable binary code of library routines.
	File 4 — Product maintenance file.

BASIC 3

Tape Label	Tape Contents	
REL8A	REL8A contains a program library in Update format. It has four files.	
	File 1	Program library for BASIC.
	File 2	Absolute binary code and compiler overlays.
	File 3	Relocatable binary code of library routines.
	File 4	Product maintenance file.

APL 2

Tape Label	Tape Contents	
REL8B	REL8B contains a program library in Modify format. It has 14 files.	
	File 1	Program library for APL.
	File 2	APLLIB. Relocatable binaries of APL.
	File 3	APLPROD. Absolute binaries of APL.
	File 4	TAPLTST. APL verification test jobs.
	File 5	TAPLOUT. Sample output for file 4.
	File 6	NEWSF. APLNEWS, news file.
	File 7	FILESYS. Workspace, file functions.
	File 8	FILES2. Workspace, file functions.
	File 9	APLNEWS. Workspace, information.
	File 10	CATALOG. Workspace, information.
	File 11	WSFNS. Workspace, general functions.
	File 12	TAPLWS. Workspace for APL verification.
	File 13	Reserved.
	File 14	Absolute binaries for APL loader.

PASCAL 170

Tape Label	Tape Contents	
REL8C	REL8C contains a program library in Update format. It has four files.	
	File 1	Program library for PASCAL 170.
	File 2	Absolute binary code for PASCAL 170.

Tape Label		Tape Contents
	File 3	The PASCAL 170 library, containing source code of the external declarations of the library routines (appearing as text on an itemize of the file), and relocatable binary code of the library routines.
	File 4	Product maintenance file.

DATA CATALOGUE 2

Tape Label		Tape Contents
REL11A	REL11A contains a program library in Update format. It has 20 files.	
	File 1	Program library for Data Catalogue.
	File 2	SEGLOAD directives.
	File 3	Absolute binary of DCUPD.
	File 4	Absolute binary of DCSEL.
	File 5	Absolute binary of DCRPT.
	File 6	Absolute binary of DCRET.
	File 7	Absolute binary of DCCONVT.
	File 8	Absolute binary of DCUTL.
	File 9	Absolute binary of DCIDX.
	File 10	Absolute binary of DCGEN.
	File 11	Absolute binary of DCCONGN.
	File 12	Relocatable binary of DCUPD.
	File 13	Relocatable binary of DCSEL.
	File 14	Relocatable binary of DCRPT.
	File 15	Relocatable binary of DCRET.
	File 16	Relocatable binary of DCCONVT.
	File 17	Relocatable binary of DCUTL.
	File 18	Relocatable binary of DCIDX.
	File 19	Relocatable binary of DCGEN.
	File 20	Relocatable binary of DCCONGN.

INFORMATION MANAGEMENT FACILITY 1

Tape Label	Tape Contents
REL11B	REL11B contains a program library in Update format. It has four files. File 1 Program library for IMF. File 2 Absolute binary code of IMF. File 3 Relocatable binary of IMFLIB library. File 4 Product maintenance file.

DATABASE UTILITIES 1

Tape Label	Tape Contents
REL11D	REL11D contains a program library in Update format. It has four files. File 1 Program library for DBU. File 2 Absolute binary code of recover/restore. File 3 Relocatable binary of logging routines. File 4 Product maintenance file.

QUERY UPDATE 3

Tape Label	Tape Contents
REL11E	REL11E contains a program library in Update format. It has four files. File 1 Program library for Query Update 3. File 2 Absolute binary code of Query Update 3. File 3 Library routines for DMSLIB. File 4 Product maintenance file.

CYBER DATABASE CONTROL SYSTEM 2

Tape Label	Tape Contents	
REL11G	REL11G contains a program library in Update format. It has four files.	
	File 1	Program library for CDCS, including utilities.
	File 2	Absolute binary code of CDCS, DBMSTRD (master directory utility), DBQRFA (quick recovery file applier utility), DBRCVR (recovery utility for reconstruct and restore operations), DBQRFI (quick recovery file initialization utility), DBREC (basic recovery utility), and CDCSBTF (batch test facility).
	File 3	Relocatable binary code of CDCS object time routines (to be loaded with the user's job).
	File 4	Product maintenance file.

DATA DESCRIPTION LANGUAGE 3

Tape Label	Tape Contents	
REL11H	REL11H contains a program library in Update format. It has four files.	
	File 1	Program library for DDL.
	File 2	Absolute binary code of DDL.
	File 3	Binary code of library routines for DDL.
	File 4	Product maintenance file.

TRANSACTION FACILITY 1

Tape Label	Tape Contents	
REL12C	REL12C contains a program library in Modify format. It has four files.	
	File 1	Program library for TAF.
	File 2	Absolute binary code of TAF and PP (1TP) code.
	File 3	Empty file.
	File 4	Product maintenance file.

INTERACTIVE FACILITY 1

Tape Label	Tape Contents
REL12E	REL12E contains a program library in Modify format. It contains four files. File 1 Program library for IAF. File 2 Absolute binary code of IAF and PP (1TN,1TO, and 1TA) code. File 3 IAF HELP file. File 4 Product maintenance file.

MESSAGE CONTROL SYSTEM 1

Tape Label	Tape Contents
REL12F	REL12F contains a program library in Update format. It contains four files. File 1 Program library for MCS and Application Definition Language Processor (ADLP). File 2 Absolute binary code of MCS and ADLP. File 3 Relocatable binary code of MCSLIB. File 4 Relocatable binary of MCS program library.

NETWORK ACCESS METHOD 1

Tape Label	Tape Contents
REL12G	REL12G contains a program library in Update format. It has four files. File 1 Program library for NAM. File 2 Absolute binary code of NAM and PP (PIP) code. File 3 Relocatable binary code of NETIO and NETIOD. File 4 Product maintenance file.

REMOTE BATCH FACILITY 1

Tape Label	Tape Contents
REL12H	REL12H contains a program library in Update format. It has four files.

File 1	Program library for RBF.
File 2	Absolute binary code of RBF and PP (1DC) code.
File 3	Relocatable binary code of RBFLIB.
File 4	Product maintenance file.

CYBER CROSS SYSTEM 1

Tape Label	Tape Contents
REL13A	REL13A contains a program library in Update format. It has 11 files.

File 1	Program library for CYBER Cross System.
File 2	Absolute binary code of CYBER Cross System.
File 3	Empty file.
File 4	Empty file.
File 5	Pascal cross-reference.
File 6	MPLINK.
File 7	MPEDIT.
File 8	CCP Pascal compiler.
File 9	CYBER 170, CYBER 70, and 6000 Computer Systems Pascal compiler (bootstrap).
File 10	Empty file.
File 11	Cross program listing.

COMMUNICATIONS CONTROL PROGRAM 3

CCP release tapes contain program libraries in Update format.

Tape Label	Tape Contents	
REL13B	REL13B contains the CCP base and base support program library. It has seven files.	
	File 1	Program library for CCP base and base support.
	File 2	Absolute binaries of the utility programs EXPAND and Autolink.
	File 3	MUX firmware (phase 1) load module.
	File 4	Dump bootstrap load module.
	File 5	System autostart (SAM) load module.
	File 6	Symbol table for dump bootstrap load module.
	File 7	Full option object file (combined binary library) for generation of phase 2 variant load modules.
	During CCP installation the system writes six additional files to REL13B.	
	File 8	Compiler listing for EXPAND and Autolink.
	File 9	CCP listing for phase 1 load module build.
	File 10	CCP listing for dump bootstrap load module build.
	File 11	CCP listing for system autostart load module build.
	File 12	CCP assembly listing for combined binary library.
	File 13	CCP Pascal listing for combined binary library.
REL13C	REL13C contains the 2550 variant. It has six files.	
	File 1	CCP variant (phase 2) load module.
	File 2	CCP PICB load module.
	File 3	Symbol table for CCP variant and PICB.
	File 4	CCP listing for CCP variant build (MPLINK).
	File 5	Listing for CCP variant build (MPEDIT).
	File 6	Listing for PICB build (PASCAL, MPLINK, and MPEDIT).

Tape Label	Tape Contents
REL13E	REL13E contains the CCP on-line diagnostics program library. It contains one file.
	File 1 Program library for on-line diagnostics.
REL13F	REL13F contains the CCP remote concentrator program library (Link Interface Program). It contains one file.
	File 1 Program library for the remote concentrator (Link Interface Program).

PL/I 1

Tape Label	Tape Contents
REL14A	REL14A contains a program library in Update format. It has four files.
	File 1 Program library for the PL/I compiler and run-time system.
	File 2 Absolute binary of the compiler.
	File 3 Relocatable binary of PLILIB library.
	File 4 Product maintenance file.

COMMON CODE GENERATOR 1

Tape Label	Tape Contents
REL14B	REL14B contains a program library in Update format. It has three files.
	File 1 Program library for Common Code Generator.
	File 2 Empty file.
	File 3 Empty file.

CYBER MASS STORAGE SUBSYSTEM 1

Tape Label	Tape Contents
REL14C	REL14C contains a program library in Modify format. It has four files. File 1 Program library for MSS. File 2 Absolute binary code of MSS. File 3 Relocatable binary code of library routines. File 4 Product maintenance file.

REMOTE HOST FACILITY AND REMOTE HOST PRODUCTS

Tape Label	Tape Contents
REL16A	REL16A contains a program library in Update format. It has four files. File 1 Program library for RHF subsystem and utilities. File 2 Absolute binary code of RHF. File 3 Relocatable binary code of RHF library. File 4 Product Maintenance File.
REL16B	REL16B contains a program library in Update format. It has four files. File 1 Program library for RHF applications (PTF, PTFS, QTF, QTFS, and MLTF). File 2 Absolute binary code for RHF applications. File 3 Empty file. File 4 Product maintenance file.

RELEASED DEADSTART TAPE TEXT RECORDS D

This appendix presents the abbreviated instruction displays for CMRDECK, IPRDECK, and APRDECK entries (figures D-1, D-2, and D-3). Press the + key at the console to alternate the first two screens of the CMRDECK and IPRDECK displays with the third.

To see the CMRDECKs, APRDECKs, IPRDECKs, and LIBDECKs on the release deadstart tape, access the SYSTEM file with the ASSIGN or COMMON command, then enter the CATALOG command and specify the T parameter (refer to NOS 2 Reference Set, Volume 3).

Further information on CMRDECK, APRDECK, and IPRDECK is in sections 7, 8, and 9, respectively. Further information on LIBDECK is in section 10. Detailed descriptions of all SYSEDIT directives are in the NOS 2 System Maintenance Reference Manual. Procedures for displaying and modifying CMRDECK and IPRDECK entries during deadstart are in the NOS 2 Operator/Analyst Handbook.

```
CMRINST
INSTRUCTIONS FOR INITIAL SETUP OF THE OPERATING SYSTEM.
USE + KEY TO SEE NEXT PAGE.

NEXT. WILL CAUSE THE LOAD TO CONTINUE.
IPR. WILL CONTINUE THE LOAD TO THE *IPRDECK* DISPLAY.
GO. WILL CONTINUE THE LOAD WITHOUT FURTHER DISPLAYS.
THE RIGHT BLANK KEY TOGGLES THE DISPLAY.

DAYFILE=1,400.  DAYFILE RESIDES ON EQ 1, CM BUFFER LENGTH = 400.
ACCOUNT=1,400.  ACCOUNT RESIDES ON EQ 1, CM BUFFER LENGTH = 400.
ERRLOG=1,100.   ERRLOG RESIDES ON EQ 1, CM BUFFER LENGTH = 100.
MAINLOG=1,400.  MAINLOG RESIDES ON EQ 1, CM BUFFER LENGTH = 400.
FOT=10.   SET FOT LENGTH = 10.
FNT=23.   SET FNT LENGTH = 23.
EJT=400.  SET EJT LENGTH = 400.
PPB=XX. NUMBER OF 102B WORD PP BUFFERS FOR 819 OR ECS TRANSFERS.
QFT=400.  SET NUMBER OF QFT ENTRIES = 400.
NCP=17.   SET THE NUMBER OF CONTROL POINTS = 17.
PPU=X,Y,...Z.  TURN OFF PPU X,Y,...,Z.  (0,1,2,10 ILLEGAL)
NAME=CCC-CCC. SET THE SYSTEM NAME = CCC-CCC.
VERSION=CCC-CCC.  SET VERSION NAME = CCC-CCC.
IPD=0.    ASSEMBLE INSTALLATION PARAMETER DECK 0.
LIB=N.    BUILD SYSTEM USING LIBDECK N (N = 0-7).
LIDT=XXX. SET LID TABLE LENGTH.
MID=MM.   SET MACHINE ID = MM (DEFAULT MNEMONIC = *AA*).
MINCM=XXXX.  SET CENTRAL MEMORY SIZE TO XXXX HUNDRED WORDS.
NAMIAF=N,H.  NUMBER OF NETWORK (N) AND HIGH SPEED (H) TERMINALS.
PRESET,N. PRESET MMF LINK DEVICE FOR *N* SHARED EQUIPMENTS.
PRESET=X,Y,...Z.  PRESET INDEPENDENT SHARED DEVICES.
LBC,CT,C1,... CN.  LOAD DISK CONTROLLERS.
     CT = CONTROLLER LOAD TYPE.
          HT = LOAD 7054 CONTROLLER.
          NH = DO NOT LOAD 7054 CONTROLLER.
          FT = LOAD 7154 CONTROLLER.
          NF = DO NOT LOAD 7154 CONTROLLER.
          FM = LOAD 7155 CONTROLLER.
          NM = DO NOT LOAD 7155 CONTROLLER.
          AD = LOAD FSC ADAPTOR.
          NA = DO NOT LOAD FSC ADAPTOR.
          PH = LOAD 7155 PARALLEL CONTROLLER.
          NP = DO NOT LOAD 7155 PARALLEL CONTROLLER.
          NN = DO NOT LOAD NAD CONTROLLER.
     CX = CHANNELS TO LOAD WITH TYPE CT.

EST DEFINITION.
EQXX=TY,ST,EN,UN,A,B,C,D,OP. DEFINE EQUIPMENT XX AS FOLLOWS
     TY = TYPE (2 LETTERS).    A - D = CHANNELS.
     N  = NUMBER OF UNITS.     OP = TAPE HARDWARE OPTION.
     ST = STATUS (ON, OFF).    FD = FLAW DECK NUMBER.
     EN = EQUIPMENT NUMBER.    HB = HASH TABLE LENGTH.
     UN = UNIT NUMBER.              (BUFFERED DISKS)
```

Figure D-1. Released CMRDECK Instructions, CMRINST (Sheet 1 of 2)

```
EQXX=YYYY.  ENTER YYYY AS OCTAL ENTRY FOR EQ XX.
EQXX=DE,ST,1000,FD.  SET ECS EQUIPMENT  1000K  (250K).
EQXX=DP,ST,4000,A,B,FD.  DDP EQUIPMENT  4000K  (1M).
EQXX=DP,ST,7777,A,B,FD.  DDP EQUIPMENT  7777K  (2M).
EQXX=DT-N,ST,EN,UN,A,B,FD,HB.  N = NUMBER OF UNITS.
EQXX=DT-NC,ST,EN,U0,,,U7,A,B,FD,HB.  NON-CONTIGUOUS FORMAT.
     DB = 885-42 FULL TRACK.     DM = 885 HALF TRACK.
     DI = 844-21 HALF TRACK.     DQ = 885 FULL TRACK.
     DJ = 844-4X HALF TRACK.     DV = 819 SINGLE DENSITY.
     DK = 844-21 FULL TRACK.     DW = 819 DOUBLE DENSITY.
     DL = 844-4X FULL TRACK.
EQXX=MT-N,ST,EN,UN,A,B,C,D,OP. ENTER CONSECUTIVE MAGNETIC TAPES.
     THE FOLLOWING ENTRIES ARE CLEARED IF EQ IS REDEFINED -
ASR=X,Y,...,Z. SET ALTERNATE SYSTEM DEVICES.
LINK=XX.   SET EQUIPMENT XX AS MMF LINK DEVICE.
MSAL,X=E1,E2,... ,EN.  SET DEVICES FOR FILE ALLOCATION.
          (X = S,B,L,D,P,R,O,I,T)
PF=XX,TY,DM,SM,FM,DN,NC.  SET PF CONTROLS FOR DEVICE XX.
PF=XX,TY,FM,NC. FOR AUXILIARY PACKS.
     APPLIES IF INITIALIZE,AL,XX. IS USED.
     TY = TYPE OF PERMANENT FILE RESIDENCE.
     DM = DEVICE MASK.
     SM = SECONDARY MASK.
     FM = FAMILY NAME (1-7 CHARACTERS).
     DN = DEVICE NUMBER.
     NC = CATALOG TRACKS (POWER OF 2 .LE. 200).
INITIALIZE,OP,X,Y,...Z.
     INITIALIZE DEVICES X,Y,...Z WITH OPTION *OP*.
     THE OPTIONS ARE AL, PF, QF, DF, AF, EF, FP, MF.
FAMILY=X.          SET X AS FAMILY DEVICE.
REMOVE=X,Y,...Z.  SET X,Y,...Z AS REMOVABLE DEVICES.
DOWN=X,Y,...Z.    LOGICALLY REMOVE EQUIPMENTS X,Y,...Z.
SHARE=X,Y,...Z.   DEFINE SHARED EQUIPMENTS FOR MMF SYSTEM.
ISHARE=X,Y,Z.     DEFINE INDEPENDENT SHARED DEVICES.
SYSTEM=X,Y,...Z. SET X,Y,...Z AS SYSTEM DEVICES. (ALL SAME TYPE)
     *MXSY* (5) SYSTEM DEVICES MAY BE DEFINED.
TEMP=X,Y,...Z.    SET X,Y,...Z FOR SYSTEM ALLOCATION OF SPACE.
XM=MID,IIII,UUUU.
     MID  = MACHINE ID.
     IIII = HIGH SPEED DISK BUFFER SPACE/1000B.
     UUUU = USER ECS SPACE/1000B.
XM=MID,IIII,UUUU,EM.    *EM* FORCES USER ECS TO UEM.
XM=MID.     CLEARS ALLOCATION FOR MACHINE *ID*.

     DEADSTART OPTIONS (EACH ENTRY TOGGLES OPTION) -
AUTOLOAD.  DISABLES AUTOLOADING OF BUFFER CONTROLLERS.
GRENADE.  SELECTS GRENADE OPERATION AFTER AUTOLOADING.
```

Figure D-1. Released CMRDECK Instructions, CMRINST (Sheet 2 of 2)

```
IPRINST
INSTALLATION PARAMETER ENTRIES.

USE + KEY TO SEE NEXT PAGE.

GO. WILL CONTINUE THE LOAD WITHOUT FURTHER DISPLAYS.
THE RIGHT BLANK KEY TOGGLES THE DISPLAY.

CERTAIN INSTALLATION PARAMETERS REFER TO THE JOB SERVICE
CLASS *SC*.  *SC* MAY BE REPLACED BY THE FOLLOWING -
     SY   SYSTEM        RB   REMOTE
     BC   BATCH         TS   TIMESHARING
     DI   DETACHED      NS   NETWORK
     MA   MAINTENANCE   CT   COMMUNICATION

DISABLE,OP.
DISABLE,SYS.
ENABLE,OP.
ENABLE,SYS,CP.
     CP   MANDATORY CONTROL POINT FOR SUBSYSTEM.
     OP   ONE OF THE FOLLOWING OPTIONS
          ENGR                       PROBE
          EXTENDED STACK PURGING     REMOVABLE PACKS
          FILE STAGING               SCP
          LOGGING                    SCRISM
          MASTER MSS                 SECONDARY USER CARDS
          MS VALIDATION              SUBCP
          PF VALIDATION              USER ECS
     SYS  ONE OF THE FOLLOWING SUBSYSTEMS
         BIO - BATCHIO                MSS - MSS SUBSYSTEM
         CDC - CDCS                   NAM - NETWORKS
         IAF - INTERACTIVE FACILITY   RBF - REMOTE BATCH FACILITY
         MAG - MAGNET                 RHF - REMOTE HOST FACILITY
         MAP - MAP SUBSYSTEM          STM - STIMULATOR
         MCS - MESSAGE CONTROL        TAF - TRANSACTION FACILITY
         SSF - SCOPE STATION FACILITY
LOCK.     SETS CONSOLE LOCK STATUS.
UNLOCK.   SETS CONSOLE UNLOCK STATUS.
DEBUG.    TOGGLES CONSOLE DEBUG STATUS.
DELAY,T1XXX,T2XXX,...,TNXXX.
          SET DELAY TIME *TN* = XXX.
     TN = AR     AUTO RECALL (MILLISECONDS)
          CR     CPU PROGRAM RECALL (MILLISECONDS)
          JS     JOB SCHEDULER (SECONDS)
          JQ     INPUT FILE SCHEDULING (SECONDS)
          MN     MINIMUM JOB SWITCH DELAY (MILLISECONDS)
          MX     MAXIMUM JOB SWITCH DELAY (MILLISECONDS)

QUEUE,SC,QT,Q1XXX,Q2XXX,...,QNXXX.
          SET QUEUE PARAMETERS *QN* = XXXX, FOR QUEUE *QT*
          FOR SERVICE CLASS *SC*.
     QT = IN   INPUT     QN = IL   ORIGINAL PRIORITY
          EX   EXECUTING      LP   LOWER BOUND FOR PRIORITY
          OT   OUTPUT         UP   UPPER BOUND FOR PRIORITY
                              IP   INITIAL PRIORITY
                              WF   WEIGHTING FACTOR
```

Figure D-2. Released IPRDECK Instructions, IPRINST (Sheet 1 of 2)

```
CSM=CC.   SET SYSTEM CHARACTER SET MODE.
     CC = 63 - 63 CHARACTER SET      64 - 64 CHARACTER SET
KEYPM=CC. SET SYSTEM KEYPUNCH MODE.
      CC = 26  026 KEYPUNCH MODE
           29  029 KEYPUNCH MODE
TCVM=CC.  SET ASSUMED MAGNETIC TAPE CONVERSION MODE.
     CC = AS   ASCII     EB   EBCDIC     US   USASI

TDEN=CC.  SET ASSUMED TAPE DENSITY ACCORDING TO CC.
     CC = LO   200      HI   556      HY   800
          HD   800      PE   1600     GE   6250

TDTR=CC.  SET ASSUMED MAGNETIC TAPE TYPE ACCORDING TO CC.
     CC = MT   SEVEN TRACK   NT   NINE TRACK

DSD,X,CCC-CCC  SET INITIAL KEYBOARD COMMAND = CCC-CCC
          RECOVERY MODE X IS SELECTED.

SERVICE,SC,P1XXX,P2XXX,...,PNXXX.
          SET JOB SERVICE PARAMETERS *PN* = XXXX, FOR
          SERVICE CLASS *SC*.
     PN = PR      INITIAL CPU PRIORITY
          CP      CPU TIME SLICE (MILLISECONDS*64)
          CM      CENTRAL MEMORY TIME SLICE (SECONDS)
          NJ      MAXIMUM NUMBER OF JOBS
          FL      MAXIMUM FL/100B FOR ANY JOB
          AM      MAXIMUM FL/100B FOR ALL JOBS
          EC      MAXIMUM ECS FL/100B FOR ANY JOB
          EM      MAXIMUM ECS FL/100B FOR ALL JOBS
          FC      NUMBER OF FILES IN CATALOG
          FS      INDIVIDUAL INDIRECT ACCESS FILE SIZE
          CS      CUMULATIVE INDIRECT ACCESS FILE SIZE
          DS      INDIVIDUAL DIRECT ACCESS FILE SIZE
          TD      SUSPENSION TIMEOUT DELAY
          TP      INITIAL PRIORITY FOR ON LINE JOBS

SRST=NN.  SET SECONDARY ROLLOUT SECTOR THRESHOLD TO NN.

CPM,N1=XX1,N2=XX2.  ALTER CP MULTIPLIER TYPE NX.
          N = 0 OR 1          XX = 1 - 6
LID=CCC,X,X,X,X.
         THIS ENTRY DEFINES THE ENTRIES IN THE LID TABLE.
         WHERE *CCC* IS A THREE CHARACTER LID AND
         *X* CAN BE EITHER, H (HOST), L(LINKED),.
         D(DISABLED) OR V(VALIDATE).
```

Figure D-2. Released IPRDECK Instructions, IPRINST (Sheet 2 of 2)

```
APRINST
INSTRUCTIONS FOR ENTERING DISK FLAWS.

BELOW IS A LIST OF ENTRIES WITH WHICH TO SET TRACK
FLAWS IN TRTS.

TRACK RESERVATION ENTRIES.

CTK.               CLEAR PREVIOUS RTK, STK AND TTK ENTRIES.
STK=NNNN.          SET RESERVATION ON LOGICAL TRACK NNNN.
TTK=NNNN.          TOGGLE RESERVATION ON LOGICAL TRACK NNNN.
RTK=AIIIIII.       RESERVE ECS BLOCK, ADDRESS AIIIIII.
TKF=CNNN,TBB,SCC.  RESERVE 819 TRACK.
RTK=CNNN,TBB,SCC.  RESERVE 819 TRACK.
    CNNN = CYLINDER NNN.
    TBB = HEAD GROUP BB.
    SCC = SECTOR CC.
```

Figure D-3. Released APRDECK Instructions, APRINST

INDEX

COMMENT SHEET

MANUAL TITLE: CDC NOS Version 2 Installation Handbook

PUBLICATION NO.: 60459320 REVISION: B

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

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

CUT ALONG LINE

AA3419 REV. 4/79 PRINTED 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

CUT ALONG LINE