



---

**NOS/BE VERSION 1  
INSTALLATION HANDBOOK**

---

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

## REVISION RECORD

REVISION	DESCRIPTION
A (11-1-75)	Original release.
B (3-5-76)	Revised to reflect PSR Summary level 420. CYBER LOADER 1.1, COMPASS 3.3, CYBER Record Manager 1.4, FORTRAN Extended 4.6, Sort/Merge 4.4, and BASIC 3.1 are updated. COBOL 5 is a new product with this release.
C (7-16-76)	Revised to reflect PSR Summary level 430. New features include 844 Factory Format, 844-41/44 Support, 844 Expander Support, Job Management and System Control Point enhancements, INTERCOM Restart, and Enhanced Station performance improvement. CCP Support Software 1 is replaced by CYBER Cross System 1.1. The operating system level is now NOS/BE 1.1; CYBER Loader is now version 1.2.
D (12-3-76)	Revised to reflect PSR Summary level 439. A new feature, Common Memory Manager, is included. Products updated are SYMPL 1.2, COBOL 4.6, Sort/Merge 4.5, CDCS 1.1, QU 3.1, DDL 2.1, DBU 1.1 and COBOL 5.1.
E (4-25-77)	Corrects various technical and typographical errors, adds information on CYBER Control Language, 67x tape units, 844 full-track recording mode, programmable format control for 580 printers, and system idle mode; and documents the NOS/BE 1.2 System at PSR level 447/446. Page numbering in this manual is changed as follows: Material formerly in the introduction is now in part I, part I material is now in part II, and part II material is now in part III. This manual obsoletes all previous editions.
F (8-1-77)	Corrects various technical and typographical errors and documents the NOS/BE 1.2 system at PSR level 454.
G (12-9-77)	Corrects various technical and typographical errors and documents the NOS/BE 1.2 system at PSR level 461. Procedures to deadstart using a 7152 Mass Storage/Magnetic Tape Controller are included.
H (6-13-78)	Corrects various technical and typographical errors and documents the NOS/BE 1.3 system at PSR level 473/470. New products included with this release are CDC CYBER Interactive Debug, PL/I, INTERCOM 5, and Communication Control INTERCOM. Products updated are UPDATE 1.3, COMPASS 3.5, CDC CYBER Loader 1.4, CMM 1.1, BAM 1.5, AAM 2.0, FCL 4.7, FORTRAN Extended 4.7, COBOL 4.7, COBOL 5.2, Sort/Merge 4.6, CDCS 1.2, CDCS 2.0, DBU 1.2, DDL 2.2, DDL 3.0, QU 3.2, SYMPL 1.3, FORM 1.1, 8-Bit Subroutines 1.1, BASIC 3.2, ALGOL 4.2, and CCP 1.1. This revision also includes information for installing NOS/BE 1.3 on a CDC CYBER 170 Model 176. This manual obsoletes publication number 60454830. Because of extensive changes to this manual, chart tape and dots are not used and all pages reflect the latest revision level. This edition obsoletes all previous editions.
J (10-20-78)	Revised to reflect PSR level 481. Features documented include the COMPASS common common decks and the new CCI installation procedure.
K (2-19-79)	Revised to reflect PSR summary level 488. Features documented include ALGOL 5, FORTRAN Data Base Facility, user direct access ECS swapping, and FNT threshold support.
L (7-20-79)	Revised to support NOS/BE 1.3, third corrective code release. Features documented include Gemini load leveling, FORTRAN 5, EXPORT High Speed, Data Catalogue 2, and the addition of direct access and actual key to AAM Extended.
Publication No. 60494300	

**REVISION LETTERS I, O, Q AND X 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.

© 1975, 1976, 1977, 1978, 1979  
by Control Data Corporation

Printed in the United States of America





# LIST OF EFFECTIVE PAGES

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

PAGE	REV	PAGE	REV	PAGE	REV	PAGE	REV	PAGE	REV
Front Cover	-	II-1-23	M	II-9-1	L	II-23-5	H	II-38-1	L
Title Page	-	II-1-24	M	II-9-2	L	II-23-6	L	II-39-1	L
ii	L	II-1-25	M	II-9-3	L	II-24-1	L	II-40-1	L
ii-a/ii-b	M	II-1-26	M	II-9-4	L	II-24-2	L	II-40-2	L
iii/iv	M	II-1-27	M	II-10-1	H	II-24-2.1/ II-24-2.2	L	II-41-1	L
v	L	II-1-28	M	II-10-2	H	II-24-3	M	Part III	
vi	M	II-1-29	M	II-11-1	M	II-24-4	M	Divider	-
vii	M	II-1-30	M	II-12-1	L	II-25-1	L	III-1-1	L
viii	M	II-1-31	M	II-12-2	L	II-25-2	H	III-1-2	L
ix	L	II-1-32	K	II-12-3	L	II-26-1	K	III-1-3	L
x	L	II-1-33	K	II-12-4	H	II-26-2	K	III-1-4	L
xi	L	II-1-34	H	II-12-5	K	II-27-1	H	III-1-5	L
Part I		II-1-35	K	II-12-6	L	II-27-2	M	III-1-6	L
Divider	-	II-1-36	H	II-12-7	L	II-27-3	H	III-1-7	L
I-1-1	L	II-1-37	K	II-12-8	H	II-27-4	M	III-1-8	L
I-1-2	H	II-1-38	K	II-12-9	H	II-27-5	H	III-1-9	L
I-1-3	M	II-1-39	H	II-12-10	H	II-28-1	L	III-1-10	L
I-1-4	H	II-1-40	K	II-12-11	H	II-28-2	L	III-1-11	L
I-1-5	M	II-1-41	M	II-12-12	H	II-28-3	L	III-1-12	L
I-1-6	M	II-1-42	K	II-12-13	H	II-28-4	L	III-1-13	L
I-1-7	M	II-1-42.1/ II-1-42.2	M	II-12-14	H	II-28-5	L	III-1-14	L
I-1-8	M	II-1-43	M	II-12-15	H	II-28-6	L	III-1-15	L
I-1-8.1/ I-1-8.2	L	II-1-44	H	II-12-16	M	II-28-7	L	III-1-16	L
I-1-9	M	II-1-45	L	II-12-17	L	II-28-8	L	III-1-17	L
I-1-10	K	II-1-46	M	II-12-18	L	II-28-9	H	III-1-18	L
I-1-11	L	II-1-47	M	II-12-19	H	II-28-10	J	III-1-19	L
I-1-12	L	II-1-48	M	II-12-20	L	II-28-11	H	III-1-20	L
I-1-13	M	II-1-49	M	II-12-21	L	II-28-12	J	III-1-21	L
Part II		II-1-50	M	II-12-22	H	II-28-13	M	III-1-22	L
Divider	-	II-1-51	M	II-12-23	H	II-28-14	K	Index-1	M
II-1-1	M	II-1-52	M	II-12-24	M	II-28-15	L	Index-2	M
II-1-2	M	II-1-53	M	II-12-25	M	II-28-16	H	Index-3	M
II-1-3	M	II-1-54	M	II-12-26	K	II-28-17	L	Index-4	M
II-1-4	L	II-1-55	M	II-12-27	H	II-28-18	L	Index-5	M
II-1-5	M	II-1-56	M	II-12-28	H	II-28-19	H	Comment	
II-1-6	M	II-1-57	M	II-13-1	L	II-28-20	H	Sheet	M
II-1-6.1/ II-1-6/.2	M	II-1-58	M	II-13-2	L	II-28-21	L	Back Cover	-
II-1-7	M	II-2-1	K	II-14-1	H	II-28-22	M		
II-1-8	L	II-3-1	L	II-15-1	L	II-28-23	H		
II-1-8.1/ II-1-8.2	M	II-3-2	H	II-15-2	H	II-28-24	K		
II-1-9	L	II-3-3	M	II-15-3	L	II-28-25	H		
II-1-10	M	II-4-1	M	II-16-1	L	II-29-1	L		
II-1-11	J	II-4-2	H	II-16-2	H	II-29-2	L		
II-1-12	J	II-4-3	L	II-17-1	H	II-29-3	L		
II-1-13	L	II-4-4	H	II-17-2	M	II-29-4	L		
II-1-14	L	II-4-5	L	II-17-3	K	II-29-5	L		
II-1-15	H	II-4-6	L	II-17-4	H	II-29-6	L		
II-1-16	L	II-5-1	H	II-17-5	H	II-29-7	L		
II-1-17	M	II-6-1	H	II-18-1	H	II-29-8	L		
II-1-18	M	II-7-1	M	II-19-1	L	II-30-1	L		
II-18.1/ II-18.2	M	II-8-1	L	II-19-2	H	II-31-1	H		
II-1-19	H	II-8-2	L	II-20-1	H	II-32-1	L		
II-1-20	M	II-8-3	H	II-21-1	L	II-33-1	L		
II-1-21	M	II-8-4	H	II-21-2	L	II-33-2	L		
II-1-22	M	II-8-5	H	II-22-1	H	II-34-1	L		
		II-8-6	H	II-23-1	H	II-35-1	L		
		II-8-7	H	II-23-2	H	II-36-1	K		
		II-8-8	H	II-23-3	M	II-37-1	K		
				II-23-4	H				



## PREFACE

This handbook provides the information required for an analyst to install the Network Operating System/Batch Environment (NOS/BE). It describes the general installation process, lists the operating system and product set release materials, and gives detailed procedures and installation parameters needed for the operating system and individual product set members.

An analyst using this handbook should be familiar with CDC® CYBER 170, CYBER 70, or 6000 series computer systems. For installation purposes, information provided in the NOS/BE Reference Manual and NOS/BE Operator's Guide is needed. To set installation parameters and maintain the operating system and product set, the analyst should have access to the UPDATE Reference Manual, COMPASS Reference Manual, and SYMPL Reference Manual.

This manual is divided into three parts.

- Part I Lists the installation and verification job decks and the options available with them; also outlines the general installation procedure and illustrates the order of installation.
- Part II Lists the release materials, detailed installation procedures and parameters, and any additional information for the operating system and each product set member.
- Part III Contains a cross-reference listing showing routines that reference installation parameters.

### RELATED PUBLICATIONS

The following manuals contain additional information on the NOS/BE Version 1 supported software and hardware.

<u>Control Data Publication</u>	<u>Publication Number</u>
ALGOL Version 4 Reference Manual	60496600
ALGOL-60 Version 5 Reference Manual	60481600
BASIC Version 3 Reference Manual	19983900
COBOL Version 4 Reference Manual	60496800
COBOL Version 5 Reference Manual	60497100
COBOL Version 4 to COBOL Version 5 Conversion Aid Version 1 Reference Manual	19265021
COBOL Version 5 Diagnostic Handbook	60482500
Common Memory Manager Version 1 Reference Manual	60499200
Communications Control Program Version 1 Reference Manual	60470000
Communications Control Program Version 1 Operator's Guide	60470100
Communications Control Program Version 1 Diagnostic Handbook	60470200
Communications Control INTERCOM Version 3 Reference Manual	60471150
COMPASS Version 3 Reference Manual	60492600
CYBER Cross System 1 Reference Manual	96836000
CYBER Cross System 1 Diagnostic Handbook	96836300
CYBER Cross System 1 PASCAL Compiler Reference Manual	96836100
CYBER Cross System 1 Macro Assembler Reference Manual	96836500
CYBER Cross System 1 Micro Assembler Reference Manual	96836400
CYBER Cross System 1 Link Editor and Library Maintenance Programs Reference Manual	60471200
CYBER Common Utilities Reference Manual	60495600
CYBER Interactive Debug Reference Manual	19265356
CYBER Loader Version 1 Reference Manual	60429800
CYBER Record Manager Basic Access Methods Version 1 Reference Manual	60495700
CYBER Record Manager Version 1 Guide for Users of COBOL Version 4	60496000
CYBER Record Manager Version 1 Guide for Users of FORTRAN Extended Version 4	60495900
CYBER Record Manager Advanced Access Methods Version 2 Reference Manual	60499300
Data Handler Version 1 Reference Manual	17322100
DDL Version 2 Reference Manual	
Volume 1 Schema Definition	60498400
Volume 2 COBOL Sub-Schema Definition	60498500
Volume 3 QUERY UPDATE Sub-Schema Definition	60498600
DDL Version 3 Reference Manual	
Volume 1 Schema Definition	60481900
Volume 2 COBOL Sub-Schema Definition	60482000
Volume 3 QUERY UPDATE Sub-Schema Definition	60482100
CYBER Database Control System Version 1 Reference Manual	60498700
CYBER Database Control System Version 2 Reference Manual	60481800
Data Base Utilities Version 1 Reference Manual	60498800
Data Catalogue 2 Version 1 Reference Manual	60483200

Control Data PublicationPublication Number

EXPORT High Speed Reference Manual	60456880
Query Update Version 3 Reference Manual	60498300
FORM Version 1 Reference Manual	60496200
FORTTRAN Extended Version 4 Reference Manual	60497800
FORTTRAN Extended Version 4 Debug User's Guide	60498000
FORTTRAN Common Library Mathematical Routines Reference Manual	60498200
FORTTRAN Data Base Facility Reference Manual	60482200
FORTTRAN Version 5 Reference Manual	60481300
FORTTRAN 4/5 Conversion Aid Reference Manual	60483000
Graphics Product Family General Information Manual	76077000
INTERCOM Version 4 Reference Manual	60494600
INTERCOM Version 4 Interactive Procedure Guide	60495200
INTERCOM Version 4 Interactive Command Summary	60495300
INTERCOM Version 4 Guide for Users of FORTRAN Extended	60495000
INTERCOM Version 4 Guide for Users of COBOL	60495100
INTERCOM Version 4 Remote Batch User's Guide	60496100
INTERCOM Version 4 Remote Batch Command Summary	60495400
INTERCOM Version 4 Multiuser Job Capability Reference Manual	60494700
INTERCOM Version 5 Reference Manual	60455010
INTERCOM Version 5 Guide for Users of FORTRAN Extended	60455950
INTERCOM Version 5 Guide for Users of COBOL	60455960
INTERCOM Version 5 Remote Batch User's Guide	60455890
INTERCOM Version 5 Interactive Command Summary	60455840
INTERCOM Version 5 Remote Batch Command Summary	60455850
INTERCOM Version 5 Multiuser Job Capability Reference Manual	60456070
NOS/BE Version 1 Reference Manual	60493800
NOS/BE Version 1 Operator's Guide	60493900
NOS/BE Version 1 Diagnostic Handbook	60494400
NOS/BE Version 1 System Programmer's Reference Manual	60494100
NOS/BE Version 1 User's Guide	60494000
NOS/BE Version 1 Applications Installation Handbook	84000980
NOS/BE Version 1 On-Line Maintenance Software Reference Manual	60453900
NOS/BE Manual Abstracts	84000470
NOS/BE Version 1 Diagnostic Index	60456490
PL/I Reference Manual	60388100
Programming Reference Aids	60158600
SCED Version 1 User's Guide	60494800
SIFT Version 1 Programming Systems Bulletin	60496500
Sort/Merge Version 4 Reference Manual	60497500
SCOPE 3.3 to SCOPE 3.4 Conversion Aids Programming Systems Bulletin	60358200
SYMPL Version 1 Reference Manual	60496400
Update Version 1 Reference Manual	60449900
8-Bit Subroutines Reference Manual	60495500
1700 MSOS IMPORT HS 1 Reference Manual	96719200
SCOPE 2.1 Operator's Guide	60455090

The NOS/BE Version 1 Station Operator's Guide/Reference Manual is replaced by the NOS/BE Version 1 Operator's Guide and the SCOPE Version 2 Operator's Guide. The NOS/BE to NOS/BE link is described in the NOS/BE Version 1 Operator's Guide. The NOS/BE to SCOPE 2 link is described in the SCOPE 2 Operator's Guide. Where the Station Operator's Guide/Reference Manual is mentioned in the text, the reader should refer to the appropriate operator's guide.

Effective with the release of PSR Level 499 (NOS/BE 1.3, third corrective code release), CEMS support for the 657 and 659 tape units and 841 multiple disk drive is withdrawn. References to these devices and their interfaces to the software have been deleted from all product support manuals except this manual. Although the installation information for these devices which are no longer supported is retained in this manual, it is not to be construed as continued CEMS support. Rather, it is an accommodation to your continued use of the equipment and the code supporting this equipment, despite the withdrawal of future CEMS support.

DISCLAIMER

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



# CONTENTS

## PART I

<p><b>1. GENERAL INSTALLATION</b> <span style="float: right;">I-1-1</span></p> <p>NOS/BE Installation Process <span style="float: right;">I-1-1</span></p> <p style="padding-left: 20px;">User Library Method <span style="float: right;">I-1-1</span></p> <p style="padding-left: 20px;">Running System Modification Method <span style="float: right;">I-1-1</span></p> <p style="padding-left: 20px;">Build Preparation <span style="float: right;">I-1-1</span></p> <p style="padding-left: 20px;">Build Techniques <span style="float: right;">I-1-2</span></p> <p style="padding-left: 20px;">User Library Techniques <span style="float: right;">I-1-2</span></p> <p style="padding-left: 40px;">Complete Builds and Assemblies <span style="float: right;">I-1-2</span></p> <p style="padding-left: 40px;">Partial Builds and Editlibs <span style="float: right;">I-1-2</span></p> <p style="padding-left: 40px;">Deadstart Tape Creation <span style="float: right;">I-1-2</span></p> <p style="padding-left: 20px;">Running System Modification Techniques <span style="float: right;">I-1-2</span></p> <p style="padding-left: 20px;">Deadstart Tape Creation and Text <span style="float: right;">I-1-3</span></p> <p style="padding-left: 40px;">Record Usage <span style="float: right;">I-1-3</span></p> <p>Product Dependencies <span style="float: right;">I-1-4</span></p> <p>Installation Deck Nomenclature <span style="float: right;">I-1-4</span></p> <p>AUTOMAT Procedure <span style="float: right;">I-1-6</span></p> <p>Obtaining Installation Jobs <span style="float: right;">I-1-7</span></p> <p>Deck Options <span style="float: right;">I-1-7</span></p> <p style="padding-left: 20px;">Common Deck Modifications <span style="float: right;">I-1-8.1</span></p> <p style="padding-left: 20px;">DST1 and DST2 Common Decks <span style="float: right;">I-1-9</span></p> <p style="padding-left: 20px;">Special Purpose Decks <span style="float: right;">I-1-10</span></p> <p style="padding-left: 40px;">MINI <span style="float: right;">I-1-10</span></p> <p style="padding-left: 40px;">ULIB <span style="float: right;">I-1-10</span></p> <p style="padding-left: 40px;">LIBS <span style="float: right;">I-1-10</span></p> <p style="padding-left: 40px;">DST1 <span style="float: right;">I-1-10</span></p> <p style="padding-left: 40px;">DST2 <span style="float: right;">I-1-10</span></p> <p style="padding-left: 40px;">DST3 <span style="float: right;">I-1-11</span></p> <p style="padding-left: 40px;">IPTEXT <span style="float: right;">I-1-11</span></p> <p style="padding-left: 40px;">COMPCOM <span style="float: right;">I-1-11</span></p> <p style="padding-left: 40px;">PLIAT <span style="float: right;">I-1-11</span></p> <p>System Texts <span style="float: right;">I-1-11</span></p>	<p style="padding-left: 40px;">Tape Security Dependencies <span style="float: right;">II-1-14</span></p> <p style="padding-left: 40px;">IP.WEC <span style="float: right;">II-1-15</span></p> <p>Miscellaneous Installation Variables <span style="float: right;">II-1-15</span></p> <p style="padding-left: 20px;">Password Definitions for the System <span style="float: right;">II-1-15</span></p> <p style="padding-left: 20px;">Dayfile <span style="float: right;">II-1-15</span></p> <p style="padding-left: 20px;">Macros to Define Spacing Code Arrays for Janus <span style="float: right;">II-1-15</span></p> <p style="padding-left: 20px;">Macro to Define the EC Parameter on the Job Statement <span style="float: right;">II-1-16</span></p> <p>CMR Configuration Parameters (CMRIP) <span style="float: right;">II-1-16</span></p> <p>Logical ID Table (LID) <span style="float: right;">II-1-18.1</span></p> <p>System Seconds <span style="float: right;">II-1-18.1</span></p> <p>Weighting Constants <span style="float: right;">II-1-19</span></p> <p>Table Structures <span style="float: right;">II-1-19</span></p> <p>Equipment Status Table <span style="float: right;">II-1-19</span></p> <p>Equipment Configuration <span style="float: right;">II-1-20</span></p> <p>Record Block Reservation Table (RBR) <span style="float: right;">II-1-24</span></p> <p>FLAW Table <span style="float: right;">II-1-25</span></p> <p>Computing Physical Addresses for Converting RB Numbers <span style="float: right;">II-1-26</span></p> <p>CMR Equipment Configuration Examples <span style="float: right;">II-1-27</span></p> <p>Deadstart Installation Parameters (DSLCOM) <span style="float: right;">II-1-29</span></p> <p>Permanent Files and Device Set Installation <span style="float: right;">II-1-31</span></p> <p>Disk Space Threshold Setting <span style="float: right;">II-1-33</span></p> <p>Scheduling Parameters <span style="float: right;">II-1-35</span></p> <p style="padding-left: 20px;">Job Classes <span style="float: right;">II-1-35</span></p> <p style="padding-left: 20px;">Central Memory <span style="float: right;">II-1-39</span></p> <p style="padding-left: 20px;">ECS Usage <span style="float: right;">II-1-40</span></p> <p style="padding-left: 20px;">CPU <span style="float: right;">II-1-40</span></p> <p>System Idle Mode <span style="float: right;">II-1-40</span></p> <p>ECS Installation Parameters (ECSCOM) <span style="float: right;">II-1-41</span></p> <p>System Circular Buffers (SCB) <span style="float: right;">II-1-42</span></p> <p>Macros to Construct ECS Label <span style="float: right;">II-1-42</span></p> <p>ECS System Segments <span style="float: right;">II-1-43</span></p> <p style="padding-left: 20px;">Segment Activity Count <span style="float: right;">II-1-43</span></p> <p style="padding-left: 20px;">Calling Sequence <span style="float: right;">II-1-43</span></p> <p>ECS as a Link Medium Between Mainframes <span style="float: right;">II-1-44</span></p> <p>Update Installation Options <span style="float: right;">II-1-45</span></p> <p>Common Memory Manager Version 1 (CMM) <span style="float: right;">II-1-45</span></p> <p>Deadstart Loading the Operating System <span style="float: right;">II-1-46</span></p> <p>Controlware Binary Creation <span style="float: right;">II-1-46</span></p> <p>Sample Job for Creation of Installation CMRs and CMR Libraries <span style="float: right;">II-1-48</span></p> <p>System Residency <span style="float: right;">II-1-49</span></p> <p>Enhanced Station Support; SPOTS (Spun Off Tasks) Core Requirements <span style="float: right;">II-1-49</span></p> <p>Accounting for the SPOTS <span style="float: right;">II-1-51</span></p> <p>Symmetric Station <span style="float: right;">II-1-51</span></p> <p>CDC CYBER Utilities <span style="float: right;">II-1-56</span></p> <p>Accounting For GENLDPF Generated Jobs <span style="float: right;">II-1-56</span></p> <p>Gemini Load Leveling <span style="float: right;">II-1-56</span></p> <p style="padding-left: 20px;">Installation Parameters <span style="float: right;">II-1-56</span></p> <p style="padding-left: 20px;">Load Leveling Algorithm <span style="float: right;">II-1-57</span></p> <p style="padding-left: 20px;">Notes and Cautions <span style="float: right;">II-1-57</span></p> <p>Deadstart Dump Analyzer <span style="float: right;">II-1-58</span></p>
-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

## PART II

<p><b>1. NOS/BE VERSION 1.4</b> <span style="float: right;">II-1-1</span></p> <p>Hardware Configuration <span style="float: right;">II-1-1</span></p> <p>Release Materials <span style="float: right;">II-1-1</span></p> <p>Notes and Cautions <span style="float: right;">II-1-2</span></p> <p>Notes and Cautions (Shared RMS Environment) <span style="float: right;">II-1-3</span></p> <p>Notes and Cautions (Tape Scheduling) <span style="float: right;">II-1-3</span></p> <p>Notes and Cautions (ATS 679 Tape Drives) <span style="float: right;">II-1-4</span></p> <p>Installation Procedures <span style="float: right;">II-1-4</span></p> <p>General Installation Parameters (IPARAMS) <span style="float: right;">II-1-4</span></p> <p>Tape Processing Installation Parameters (CIOCUM) <span style="float: right;">II-1-10</span></p> <p style="padding-left: 20px;">Automatic Assignment Bit <span style="float: right;">II-1-13</span></p> <p style="padding-left: 20px;">Automatic Scratch Status <span style="float: right;">II-1-13</span></p> <p style="padding-left: 20px;">Prestaging Bits <span style="float: right;">II-1-13</span></p> <p style="padding-left: 40px;">Unit Reservation <span style="float: right;">II-1-13</span></p> <p style="padding-left: 40px;">Prestaging <span style="float: right;">II-1-13</span></p> <p>Overcommitment Bit <span style="float: right;">II-1-14</span></p> <p>Miscellaneous Bits <span style="float: right;">II-1-14</span></p> <p style="padding-left: 20px;">Two Label Processors <span style="float: right;">II-1-14</span></p> <p style="padding-left: 20px;">EOJ Tape Unload <span style="float: right;">II-1-14</span></p> <p style="padding-left: 20px;">Operator Cannot Override VSN <span style="float: right;">II-1-14</span></p> <p style="padding-left: 20px;">Label Debug <span style="float: right;">II-1-14</span></p> <p style="padding-left: 20px;">Option Dependencies (IP.TSG) <span style="float: right;">II-1-14</span></p> <p style="padding-left: 40px;">Auto Assign Dependencies <span style="float: right;">II-1-14</span></p> <p style="padding-left: 40px;">Prestage and Overcommitment Dependencies <span style="float: right;">II-1-14</span></p>	<p><b>2. COMPASS VERSION 3.6 AND COMMON COMMON DECKS</b> <span style="float: right;">II-2-1</span></p> <p>Release Description <span style="float: right;">II-2-1</span></p> <p>Release Materials <span style="float: right;">II-2-1</span></p> <p>General Description <span style="float: right;">II-2-1</span></p> <p>Installation Parameter <span style="float: right;">II-2-1</span></p> <p>Installation Procedures <span style="float: right;">II-2-1</span></p>
-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

3. CDC CYBER RECORD MANAGER 1 BASIC ACCESS METHODS	II-3-1	Data Reduction Phase	II-8-6
		Miscellaneous Information	II-8-7
		Type Time	II-8-8
		Changes to INTERCOM	II-8-8
Release Description	II-3-1		
Notes and Cautions	II-3-1		
Additional Information	II-3-1	9. FORTRAN EXTENDED VERSION 4	II-9-1
Initialization Modules	II-3-1		
Sequential and Word Addressable I/O Modules	II-3-1	Release Materials	II-9-1
Installation Parameters	II-3-3	Limitations	II-9-1
Installation Procedures	II-3-3	Compiler Installation Parameters	II-9-1
		Compiler Program Library Structure	II-9-2
		TSTEXT	II-9-2
		FTNMAC	II-9-2
		FTNTEXT	II-9-2
		(0,0) Overlay (FTN)	II-9-3
		TS Mode Overlay	II-9-3
		(1,0) Overlay (FTN10)	II-9-3
		Optimizing and Debugging Mode Overlays	II-9-3
		(2,0) Overlay (FTN20)	II-9-3
		(2,3) Overlay (FTN23)	II-9-3
		(2,1) Overlay (FTN21)	II-9-3
		(2,2) Overlay (FTN22)	II-9-3
		(2,5) Overlay (FTN25)	II-9-3
		(2,4) Overlay (FTN24)	II-9-3
		Installation Instructions	II-9-4
4. CRM ADVANCED ACCESS METHODS 2	II-4-1		
Release Description	II-4-1	10. COBOL VERSION 4.7	II-10-1
AAM Initial	II-4-1		
AAM Extended	II-4-1	Release Materials	II-10-1
Release Materials	II-4-1	Limitations	II-10-1
Limitations	II-4-2	Installation Parameters	II-10-1
AAM Initial Installation Parameters	II-4-2	Special Installation Procedures	II-10-1
AAM Initial Installation Parameters	II-4-3	Installation Procedures	II-10-2
User Additions to AAM Extended	II-4-3		
Installation Procedures for AAM Initial	II-4-5	11. SORT/MERGE VERSION 4.6	II-11-1
Installation Procedures for AAM Extended	II-4-6		
Verification Programs	II-4-6	Release Materials	II-11-1
		Hardware Configuration	II-11-1
		General Description	II-11-1
		Installation Parameters	II-11-1
		Installation Procedure	II-11-1
5. 8-BIT SUBROUTINES VERSION 1.1	II-5-1		
Release Description	II-5-1	12. INTERCOM VERSION 4	II-12-1
Release Materials	II-5-1		
Hardware Configuration	II-5-1	Release Description	II-12-1
General Description	II-5-1	Hardware Configuration	II-12-1
Installation Procedures	II-5-1	Required Hardware Options	II-12-1
		Hardware Options	II-12-1
		Release Materials	II-12-2
		Notes and Cautions	II-12-2
		Mode 4 Sequence Bit Processing in 1M1	II-12-2
		Line Skipping on 714 Nonimpact Printer	II-12-2
		Error Processing on the 711/714	II-12-2
		Character Set Support	II-12-2
		2XPPU Speed (CDC CYBER 170 Only)	II-12-2
		Limitations and System Considerations	II-12-3
		General Procedures	II-12-3
		Installation Parameters	II-12-3
		INTERCOM Common Deck Settings	II-12-3
		NOS/BE IPARAMS Settings	II-12-7
		EST Entry	II-12-7
		Configuration Parameters (Internal to CMR)	II-12-8
		CMR Multiplexer Table	II-12-8
		CMR Multiplexer Subtable General Format	II-12-8
		Multiplexer Definition Entries	II-12-9
		Port Definition Entries	II-12-9
		Line Speeds and Hard-Wired Terminals	II-12-10
		Multidrop Lines	II-12-10
		Multistation Terminals	II-12-11
		Port Distributions for Low Speed Multiplexers	II-12-11
		Driver Type Selection	II-12-12
6. FORM VERSION 1.1	II-6-1		
General Description	II-6-1		
Release Materials	II-6-1		
Hardware Configuration	II-6-1		
Installation Parameters	II-6-1		
7. ON-LINE MAINTENANCE SOFTWARE	II-7-1		
Release Description	II-7-1		
Release Materials	II-7-1		
Installing Diagnostic Programs	II-7-1		
Installation Parameters	II-7-1		
Installation Procedures	II-7-1		
8. MAINTENANCE PACKAGE	II-8-1		
Release Description	II-8-1		
SYMPL	II-8-1		
Installation Parameters	II-8-1		
Installation Procedures	II-8-1		
Verification Program	II-8-1		
Conversion Aids	II-8-1		
Installation Procedures	II-8-2		
Usage of Conversion Aids	II-8-2		
DOCK Usage	II-8-2		
STIMULATOR Operating Instructions:	II-8-3		
Automatic Table Setting	II-8-3		
Manual Table Setting	II-8-4		
Running Two Multiplexers	II-8-4		
Bringing Up INTERCOM and the STIMULATOR	II-8-4		
Description of STIMULATOR Parameters	II-8-5		
STIMULATOR Input Statement Format	II-8-5		

Command Table Structure (ICI Overlay 2CS Common Deck COMTBL)	II-12-13	18. CDC CYBER DATABASE CONTROL SYSTEM 1.2	II-18-1
COM2CC Macro	II-12-13	General Description	II-18-1
MUJ Macro	II-12-14	Release Description	II-18-1
COM1LX Macro	II-12-14	Release Materials	II-18-1
COM1LX Notes	II-12-14	Installation Procedure	II-18-1
REMOTE Macro	II-12-14		
MUJ Table Structure (IQP)	II-12-14	19. QUERY UPDATE VERSION 3	II-19-1
TBL Assembly Options	II-12-15	Release Materials	II-19-1
Table Changes and Release Settings	II-12-15	Hardware Configuration	II-19-1
EDITOR Installation Parameters	II-12-17	Installation Parameters	II-19-1
EDITOR Debug Code	II-12-22	Notes and Cautions	II-19-1
Multiuser Job Installation Parameters	II-12-22	Installation Requirements	II-19-1
Installation Procedures	II-12-23	Installation Procedure	II-19-2
LCC Programs	II-12-23		
NPU Programs	II-12-24	20. DATA DESCRIPTION LANGUAGE (DDL) VERSION 2.2	II-20-1
MUJ System Errors	II-12-24	Release Description	II-20-1
Password File Creation	II-12-25	Hardware Configuration	II-20-1
SCED Installation Parameters	II-12-27	Installation Procedure	II-20-1
Verification Procedure	II-12-27		
		21. BASIC VERSION 3	II-21-1
13. ALGOL 60 VERSION 4	II-13-1	Release Materials	II-21-1
Release Description	II-13-1	Hardware Configuration	II-21-1
Release Materials	II-13-1	Notes and Cautions	II-21-1
Limitations	II-13-1	Installation Parameters	II-21-1
Installation Parameters	II-13-1	Installation Procedures	II-21-2
Installation Procedures	II-13-2	Verification Program	II-21-2
Verification Program	II-13-2		
		22. DATA BASE UTILITIES 1.2	II-22-1
14. CDC CYBER 70 CONVERSION AIDS SYSTEM	II-14-1	Release Description	II-22-1
		Release Materials	II-22-1
15. CDC CYBER CROSS SYSTEM VERSION 1	II-15-1	Installation Procedure	II-22-1
Release Description	II-15-1	Limitations	II-22-1
Hardware Configuration	II-15-1		
Release Materials	II-15-1	23. COMMUNICATIONS CONTROL PROGRAM 1.1	II-23-1
PASCAL Organization	II-15-2	Release Description	II-23-1
PASCAL Compiler Symbol Table Page Size	II-15-3	Release Materials	II-23-1
PASCAL Cross Reference Program Table Size	II-15-3	Hardware Configuration	II-23-2
Installation Procedure	II-15-3	CCP Hardware Requirements	II-23-2
		Notes and Cautions	II-23-2
16. CDC CYBER LOADER 1	II-16-1	Installation Parameters	II-23-2
Release Description	II-16-1	Installation Procedures	II-23-4
Release Materials	II-16-1	Configuration Aids	II-23-4
General Description	II-16-1	Memory Space Requirements	II-23-4
Installation Parameters	II-16-1	Buffer Pool Selection	II-23-5
Installation Procedures	II-16-2	Installation Procedures	II-23-5
		Corrective Code	II-23-6
17. FACTORY FORMAT SUPPORT (844-21 AND 844-41)	II-17-1	Verification Programs	II-23-6
Release Materials	II-17-1	System Generation	II-23-6
General and Operational Description	II-17-1	On-Line System	II-23-6
Flaws	II-17-1		
Disk Pack Condition	II-17-1	24. COBOL 5	II-24-1
Installation Information	II-17-2	Release Materials	II-24-1
Format Utility	II-17-2	Limitations	II-24-1
Call Format and Parameter Options	II-17-2	Installation Parameters	II-24-1
Input Formats	II-17-3	Compiler Program Library Structure	II-24-2
Output Formats	II-17-3	Installation Instructions	II-24-3
Operator Intervention and Console Messages	II-17-4		
Dayfile Messages	II-17-4		
Installation Procedures	II-17-5		

25. COBOL 4 TO COBOL 5 CONVERSION AID	II-25-1	29. COMMUNICATIONS CONTROL INTERCOM 3.0	II-29-1
Release Materials	II-25-1	Release Description	II-29-1
Hardware Configuration	II-25-1	Installation Overview	II-29-2
General Description	II-25-1	Release Materials	II-29-2
Installation Procedure	II-25-1	Hardware Configuration	II-29-4
Installation Parameters	II-25-1	CCI Hardware Requirements	II-29-4
		Notes and Cautions	II-29-4
26. CDC CYBER CONTROL LANGUAGE	II-26-1	Installation Parameters	II-29-4
Release Materials	II-26-1	Type 1 Statements	II-29-5
Hardware Configuration	II-26-1	Type 2 Statements	II-29-6
General Description	II-26-1	Type 3 Statements	II-29-6
Notes and Cautions	II-26-1	Configuration Aids	II-29-7
Installation Parameters	II-26-1	Installation Procedures	II-29-7
Installation Procedure	II-26-2	2550-2 or 2551-1 Installation (PL99A)	II-29-7
		Installation Deck PL99E	II-29-8
27. INSTALLATION PROCEDURE FOR CDC CYBER 171 CONFIGURATION	II-27-1	Corrective Code	II-29-8
Required Materials	II-27-1	Verification Programs	II-29-8
Restrictions	II-27-1	System Generation	II-29-8
Required Macros	II-27-1	On-Line System	II-29-8
AUTO Macro	II-27-1		
MUXCR Macro	II-27-2	30. CDC CYBER DATABASE CONTROL SYSTEM 2	II-30-1
Installation Procedures for INTERCOM 4	II-27-2	Hardware Configuration	II-30-1
Installation Procedures for INTERCOM 5	II-27-4	Release Materials	II-30-1
		Installation Requirements	II-30-1
28. INTERCOM VERSION 5	II-28-1	Installation Procedure	II-30-1
Release Description	II-28-1	Verification Programs	II-30-1
Hardware Configuration	II-28-1		
Required Hardware Options	II-28-1	31. DATA DESCRIPTION LANGUAGE (DDL) VERSION 3.0	II-31-1
Hardware Options	II-28-1	Release Materials	II-31-1
Release Materials	II-28-2	Hardware Configuration	II-31-1
Notes and Cautions	II-28-2	Installation Procedure	II-31-1
Line/Skipping on 714 Nonimpact Printer	II-28-2		
Error Processing on the 711/714	II-28-2	32. CDC CYBER INTERACTIVE DEBUG 1	II-32-1
Character Set Support	II-28-2	Hardware Configuration	II-32-1
Limitations and System Considerations	II-28-2	Release Materials	II-32-1
General Procedures	II-28-2	Installation Parameters	II-32-1
Installation Parameters	II-28-3	Installation Procedures	II-32-1
INTERCOM Common Deck Settings	II-28-3		
NOS/BE IPARAMS Settings	II-28-6	33. PL/I 1	II-33-1
NOS/BE PPSYS Settings	II-28-6	Release Materials	II-33-1
EST Entry	II-28-7	Requirements	II-33-1
Configuration Parameters (Internal to CMR)	II-28-7	Installation Options	II-33-1
CMR Multiplexer Table	II-28-7	Program Library Structure	II-33-2
CMR Multiplexer Subtable General Format	II-28-7	Installation Decks	II-33-2
Multiplexer Definition Entries	II-28-8		
Port Definition Entries	II-28-8	34. FORTRAN COMMON LIBRARY (FCL) VERSION 4	II-34-1
Macro Definitions	II-28-8	Release Materials	II-34-1
Command Table Structure (1CI Overlay 2CS; Common Deck COMTBL)	II-28-10	Installation Options	II-34-1
COM2CC Macro	II-28-11	Program Library Structure	II-34-1
MUJ Macro	II-28-11	Installation Decks	II-34-1
EXPCOM Macro	II-28-11		
REMOTE Macro	II-28-11		
MUJ Table Structure (1QP)	II-28-12		
TBL Assembly Options	II-28-12		
Table Changes and Release Settings	II-28-12		
EDITOR Installation Parameters	II-28-14		
EDITOR Debug Code	II-28-20		
Multiuser Job Installation Parameters	II-28-20		
Installation Procedures	II-28-21		
NPU Programs	II-28-21		
MUJ System Errors	II-28-22		
Password File Creation	II-28-23		
SCED Installation Parameters	II-28-24		
Verification Procedure	II-28-25		

35. FORTRAN DATA BASE FACILITY	II-35-1	39. FORTRAN COMMON LIBRARY (FCL) VERSION 5	II-39-1
Release Materials	II-35-1	Release Materials	II-39-1
Hardware Configuration	II-35-1	Installation Options	II-39-1
Notes and Cautions	II-35-1	Program Library Structure	II-39-1
Installation Procedure	II-35-1	Installation Procedures	II-39-1
Verification Program	II-35-1		
36. ALGOL 60 VERSION 5	II-36-1	40. EXPORT HIGH SPEED	II-40-1
Release Materials	II-36-1	Release Materials	II-40-1
Limitations	II-36-1	Hardware Configuration	II-40-1
Installation Parameters	II-36-1	Limitations	II-40-1
Installation Procedures	II-36-1	Installation Parameters	II-40-1
Verification Program	II-36-1	Equipment Status Table	II-40-2
		CMR Multiplexer Table	II-40-2
		Installation Procedure	II-40-2
37. FORTRAN 4 TO FORTRAN 5 CONVERSION AID	II-37-1	41. DATA CATALOGUE 2 VERSION 1	II-41-1
Release Materials	II-37-1	Release Materials	II-41-1
Hardware Requirements	II-37-1	Installation Procedure	II-41-1
Installation Procedure	II-37-1		
Verification Program	II-37-1		
38. FORTRAN VERSION 5	II-38-1		
Release Materials	II-38-1		
Limitations	II-38-1		
Installation Parameters	II-38-1		
Installation Procedure	II-38-1		

Part III

1. CROSS REFERENCE OF INSTALLATION PARAMETERS	III-1-1
--------------------------------------------------	---------

**INDEX**

**FIGURES**

I-1-1	Installation Sequence	I-1-6	II-28-2	1QP Release Values and Update Identifiers	II-28-13
II-1-1	Batch/INTERCOM Considerations	II-1-36			
II-28-1	2CS Release Values and Update Identifiers	II-28-13			

**TABLES**

I-1-1	Installation Requirements	I-1-5	II-12-3	EDITOR Dependencies	II-12-22
I-1-2	Common Decks and System Texts	I-1-12	II-28-1	EDITOR Installation Parameters	II-28-16
II-1-1	Prohibited Channels	II-1-47	II-28-2	EDITOR Dependencies	II-28-19
II-12-1	EDITOR Installation Parameters	II-12-9	II-28-3	EDITOR Array Dimensions in IPFTN	II-28-20
II-12-2	EDITOR Array Dimensions in IPFTN	II-12-21	II-29-1	CCI Module Requirements	II-29-1



**PART I**





## NOS/BE INSTALLATION PROCESS

Installation of the NOS/BE system is an incremental process in which each product binary file is entered into a set of user library permanent files or into the host NOS/BE operating system through the use of the EDITLIB utility. This is an ordered process in which strict adherence to product interdependencies must be maintained. Several build options allow integration of the products supported by NOS/BE and onsite tailoring of these products. Two generalized methods for producing an integrated system, the user library method and the running system modification method, are provided.

### USER LIBRARY METHOD

The user library method is the most reliable and efficient method for producing a tailored NOS/BE system. This installation method maintains each product binary file in a set of user library permanent files on either a private or public device set. This set of user libraries is self-contained such that the host NOS/BE operating system supplies only the basic operational capabilities of file manipulation, job scheduling, and resource management. The user library method has the advantages of not interrupting normal batch and interactive processing in a production environment, and providing more reliable recovery from possible failures of the host NOS/BE system.

Disadvantages of this method are that certain types of build errors are not discovered until a deadstart tape has been created, and some host system modifications may be required when a key interface or a build tool such as EDITLIB changes. This method is used to build all intermediate development systems.

### RUNNING SYSTEM MODIFICATION METHOD

With the running system modification method, the installation enters each product binary file into the host NOS/BE system either from a release tape or from the reassembled binaries produced by a previously executed assembly deck. This method enables the installation to verify the integrity of these new binaries by executing verification programs as soon as the new binaries are installed, and also conserves mass storage space.

Disadvantages of this method are that a production environment cannot be maintained during the build process without frequent interruptions, and recovery of the modified NOS/BE host system may be impossible.

### BUILD PREPARATION

Installation of a NOS/BE system requires a working knowledge of the utilities Update and EDITLIB. Familiarity with the operational characteristics of the NOS/BE operating system, its constituent texts, and control language statements is assumed. Additionally, knowledge of the functional characteristics of COMPASS, CDC CYBER Loader, COPYL, ITEMIZE, FORTRAN Extended 4, FORTRAN Common Library 4, SYMPL, and CDC CYBER Control Language is required for installation of the basic operating system components. Refer to later sections of this manual for dependencies which require knowledge of other system components.

In addition to the information contained in this document, supplemental information related to a specific release level is contained in the Software Release Bulletin associated with that release level. Deck REASON on the installation decks oldpl also contains specific release installation information. The installation decks oldpl is present as file three of the batched corrective code tape. Part III of this document contains a cross-reference listing to aid in the examination of generalized installation parameters which may be tailored to a specific configuration. Submit a job of the following form to obtain a listing of the installation decks oldpl for use as a reference during the installation process, and to save these decks as a permanent file for subsequent use.

```
DECKS, T100, MT1.          (or NT1)
REQUEST (OLDPL, HY)        (or HD)          Assign BCC tape
REQUEST (NEWPL, *PF)
SKIPF (OLDPL, 2, 17)
UPDATE (F, N, *=-, C=0, R, L=A7)
CATALOG (NEWPL, DECKS, ID=INSTALL)
6/7/8/9
```

## BUILD TECHNIQUES

The key deck in either the user library or running system method of building systems is the deadstart creation deck. DST1 is used for the user library method; DST2 is used for the running system method. Inspection of the conditional UPDATE =IF DEF directives in DST1 and DST2, and the source statements they govern, provides information regarding the eventual content and library structure of the deadstart tape to be created. Deck REASON provides a shortened list of these definable parameters as well as all definable parameters for other decks. From these references, a generalized set of =DEFINE UPDATE directives should be created which reflect all of the parameters associated with the products to be installed. Each deck extracted from the installation decks oldpl should be extracted with this generalized set of defines.

UPDATE common decks are implemented to provide constant locations for installation deck modifications. This implementation is generalized as much as possible to provide a basis for a CDC CYBER Control Language procedure file to be used for deck extraction. (Examples of this usage are contained in deck REASON.) Specialized common decks which provide locations in nonrepeatable sequences, such as in DST1 and DST2, are documented later in this section.

## USER LIBRARY TECHNIQUES

The deck LIBS establishes the permanent file environment for the user library process. All user libraries are created with a dummy routine ZZZ in each library. Subsequent installation decks delete this dummy routine. The UPDATE directive =DEFINE ULIB must be added to the generalized set of defines being used when installing with the user library method.

### Complete Builds and Assemblies

All appropriate I suffixed decks are extracted and submitted for execution. All product dependencies must be honored.

### Partial Builds and Editlibs

Any products for which there is no corrective code can be entered into the user libraries by first executing the appropriate O (Overlay) suffixed deck, if one exists. (Each product which has absolute binary modules should have a corresponding O suffixed deck.) The output tape from the O deck is then assigned to the corresponding E suffixed deck to enter the new binaries into the user libraries. For products containing no overlays, the E suffixed deck is the only required deck. The NOS/BE system contained on PL1A and PL1B, as well as INTERCOM 4 and INTERCOM 5, require that the I deck be run. Some products provide special partial assembly or variant decks. These decks should be carefully examined and understood before using them.

As in complete builds, all product dependencies must be honored.

### Deadstart Tape Creation

Upon completion of execution of the last product installation deck, all user libraries should be saved on tape using a mode I DUMPF. Before executing the DST1 deadstart tape creation deck, the utility deck ULIB must be executed. This deck creates a sequential library of the libraries USERPP and USERPS. It merges user library USERCC with library USERNUC and invalidates USERNUC for subsequent rebuilds, if necessary. (The SAVE define can be used to create a second cycle of USERNUC and avoid invalidating the original library. If this option is used, the high cycles of USERPP, USERPS, and USERNUC must be purged before any rebuilding.)

Several alternative file locations for controlware, CMR libraries, CMRs, and diagnostic sequencer text records exist in DST1. These options, along with alternatives for system residency, require that DST1 be tailored to meet specific needs.

## RUNNING SYSTEM MODIFICATION TECHNIQUES

After each appropriate I suffixed deck has completed execution, the corresponding E suffixed deck must be executed. Some products do not have a corresponding E deck; for these products the binaries are added to the new system by the DST2 deadstart tape creation deck. Where an E deck exists, the output tape from each I deck must be assigned as input to the corresponding E deck. All product dependencies must be honored including completion of the E decks before the installation of dependent products. A few intermediate system EDITLIBs occur during execution of the I decks, primarily to upgrade system text files. Partial builds using the O decks should be avoided, since the frequency of their use is rather low on system levels different from the system level being built. This is true for partial assembly and variant decks also.

## DEADSTART TAPE CREATION AND TEXT RECORD USAGE

An example of how to capture the host NOS/BE system and its constituent libraries with deadstart diagnostic sequencer routines is as follows.

```
CAPTURE,IO0,T0,NT1.
ACCOUNT(local accounting information)
REQUEST(NEWSYS,PE,RING,VSU=1234)
EDITLIB(SYSTEM,ERROR=3,MSGL=1)
UNLOAD(NEWSYS)
REWIND(OUTPUT)
7/8/9
READY(NEWSYS,NEW)
REWIND(NEWSYS)
TRANS77(IPL+OSB,SYSTEM)
TRANSFER(CED+MDR,SYSTEM)
TRANSFER (1,SYSTEM)
TRANSFER(COM+LFP,SYSTEM)
TRANSFER(1,SYSTEM)
TRANSFER(OSY,SYSTEM)
TRANSFER(OSZ,SYSTEM)
TRANSFER(OMT,SYSTEM)
TRANSFER(OSJ,SYSTEM)
TRANSFER(*,SYSTEM)
INCLUDEP(SYSTEM)
INCLUDE(NUCLEUS,SYSTEM,CM)
INCLUDE(SYSOVL,SYSTEM,CM)
INCLUDE(BAMLIB,SYSTEM,DS)
INCLUDE(SYMLIB,SYSTEM,DS)
INCLUDE(FORTRAN,SYSTEM,DS)
```

(Assuming only one CMR is present.)

(Applicable for 819 driver only)	Optional
(844 half-tracking controlware)	Optional
(844 full-tracking controlware)	Optional
(MTS controlware-66x tapes)	Optional
(885 disk drive controlware)	Optional
(This directive is invalid if an attempt is made to omit an optional host system controlware package. Some combination of the above optional controlware is required by most sites.)	

(An INCLUDE directive is required for each system library to be present on the new deadstart tape)

```
COMPLETE.
ENDRUN.
7/8/9
6/7/8/9
```

An EDITLIB LISTLNT of the host NOS/BE system or an ITEMIZE listing of the host deadstart tape is required to construct the correct sequence described. Additionally, the controlware packages and optional product libraries required for the target machine configuration must be determined. An examination of the deadstart tape creation decks supplied on the installation decks oldpl can serve as a guide to the creation of this capture deck. A simplified capture deck, named DST3, is provided on the installation decks oldpl.

## PRODUCT DEPENDENCIES

When the installation of one product requires the output tape from another product as its input, that tape must be upgraded to the level of the system being built. Some products are stacked on a single release tape such that care must be taken to ensure the integrity of the final output tape during the installation process. For example, the output tape from SYMPL must be assigned as input to the maintenance package, and the output tape from 8-Bit Subroutines must be assigned as input to FORM. For those sites with available disk space, the UPDATE directive =DF SAVE will create intermediate permanent files to automatically satisfy these dependencies.

Several types of product interdependencies determine the installation order of the NOS/BE system. These dependency types are as follows.

1. Compilers, assemblers, and utilities required by subsequent products
2. Relocatable binaries required for subsequent absolute module formation
3. Assembled texts and loaders required by subsequent products
4. COMPASS XTEXT relationships
5. Relationships such as multiple oldpl source dependencies

Table I-1-1 describes relationships 1 and 2. Figure I-1-1 shows the recommended installation sequence, considering all the relationships previously described.

The flow in figure I-1-1 is optimized for processing as many decks simultaneously as possible for efficient machine utilization. Products must be installed in a top down sequence with products on the same horizontal level or within a box capable of being installed simultaneously. Every product is depicted in the chart even though many products are optional and some are mutually exclusive. Requirements for a product are described in table I-1-1 for quick reference, and in detail in part II of this document.

## INSTALLATION DECK NOMENCLATURE

The decks which correspond to products identified in figure I-1-1 and which are contained on the installation decks oldpl have the following nomenclature.

### PLnujx

The characters in the deck name indicate the following.

- PL Prefix indicating installation or verification job
- n PL number to which the job applies
- u Optional unique character for the PL number
- j Job type identifier. The following identifiers are used.
  - I Installation job
  - Im Partial assembly or variant deck; m denotes an execution sequence number.
  - E EDITLIB job
  - O Overlay reformation job
  - T Text redefinition job
  - C Catalog job
  - V Verification job
- x Optional unique character for job type identifier

The applicability of these jobs is described in the part II sections of this document that detail the installation procedures for the operating system and each product set member.

The verification jobs contained on the installation deck PL are provided so that following completion of the installation processes and the generation of a deadstart tape, the new system may be deadstarted and subjected to a validation exercise by selecting and executing these jobs. They are included to provide a quick verification so that, at a minimum, the skeleton of the whole system is in place. They are not intended to provide a comprehensive test.

TABLE I-1-1. INSTALLATION REQUIREMENTS

Product to Be Installed	Required in the Running System or User Library														
	COM-PASS	CRM Basic	CRM Advanced	FORTTRAN Extended 4 Compiler	FORTTRAN Library 4	SYMPL	Sort/Merge	DDL 2	CYBER Cross System	DBU	CCG	8 Bit	DDL 3	CCI	COBOL 5
NOS/BE (PL1A)	I														
NOS/BE (PL1B)	I														
UPDATE (PL1C)	I														
CMM (PL1C)	I														
Enhanced Station (PL1D)	I	I		I <sup>1</sup>	I <sup>1</sup>	I	I <sup>1</sup>								
CYBER Loader (PL1E)	I														
844 FORMAT (PL1F)	I														
819 Subsystem (PL1G)	I														
Shared RMS (PL1H)	I														
Gemini (PL1H)	I														
COMPASS (PL2)	I														
CRM Basic (PL3A)	I					I									
CRM Advanced (PL3B) Initial	I	B		I	I,O	I	I,O								
CRM Advanced (PL3C) Extended	I	B		I	I,O	I	I,O								
8-Bit Subroutines (PL4)	I	E			E										
FORM (PL4)	I	B	O		E	I						E			
On-Line Maintenance Software (PL5A)	I	I		I	I										
CTI (PL5B)	I														
SYMPL (PL6)	I				I <sup>2</sup>	I									
Maintenance Package (PL6)	I	I	I	I	I										
FORTTRAN Extended 4 Compiler (PL7 or PL7A)	I														
FORTTRAN Library 4 (PL8)															
Math part	I														
Remaining FCL	I	B		I			O								
COBOL (PL9)	I	B	O		O		O	I <sup>3</sup>				O			
Sort/Merge (PL10)	I	B													
INTERCOM 4 (PL12)	I	I		I	I										
INTERCOM 5 (PL14)	I	I		I	I									E	
ALGOL 4 (PL32)	I	B				B									
CYBER Cross System (PL50)	I	I		I	I										
CDCS 1 (PL54)	I	B	E					E	O						
QU 3 (PL55)	I	B	E		I		B	I,O <sup>4</sup>	I			I,O <sup>4</sup>			
DDL 2 (PL56)	I	B													
BASIC 3 (PL57)	I														
DBU (PL58)	I	B	E		I		B								
COBOL 5 (PL60)	I	B	O		O		I						I <sup>5</sup>		
CCP (PL61)															
FORTTRAN 5 Compiler (PL63)	I								I						
FORTTRAN 5 Library (PL64)															
Math part	I														
Remaining FCL	I	B		I	I		O								
FORTTRAN 4/5															
Conversion Aid (PL65)	I	I	I	I	I										
FDBF (PL66)	I	B	B		I	B	B						I		
COBOL 4/5 Conversion Aids (PL69)	I	I	I	I	I										
CCL (PL70)	I														
ALGOL 5 (PL72)	I	B		I		B									
Data Catalogue 2 (PL73)		B	E												
CDCS 2 (PL74)	I	B	B		I	B	B								B
DDL 3 (PL77)	I	B											I		
PL/I (PL79)	I	B	O	I	E	I									
EXPORT HS (PL80)	I											I			
CID (PL82)	I			I	B	I									
CCG (PL83)															
CCI (PL99)															

Legend: I = Install, E = Execute, B = Install and Execute, O = Optional at Execute Time. All products require Update to install and NOS/BE to install and execute. All product set installation jobs that run after the CDC CYBER Loader is installed use the new loader for overlay and capsule generation.

- Notes:
1. Require for installation of enhanced station if ES4XREF is defined.
  2. Math only.
  3. COBOL 4 requires DDL 2 to install if the CDCS interface is selected.
  4. QU 3 can be used with DDL 2 or DDL 3.
  5. COBOL 5 requires either DDL 2 or DDL 3 (selected at install time) if the CDCS interface is selected.

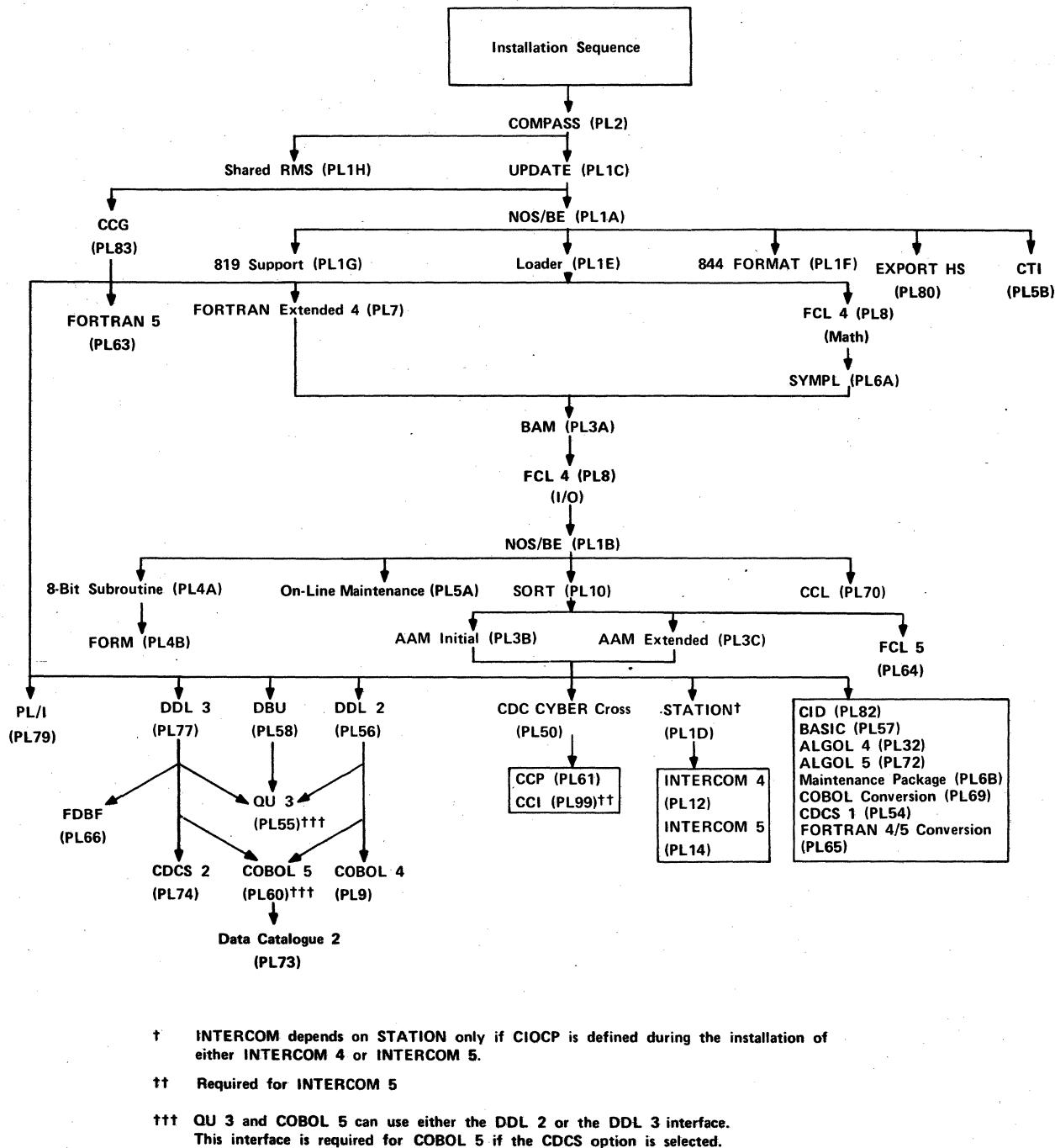


Figure I-1-1. Installation Sequence

### AUTOMAT PROCEDURE

The AUTOMAT procedure makes it easier to extract installation jobs from the installation deck PL. It produces the selected deck on a local file. The following job catalogs the AUTOMAT procedure.

```

AUTOM,T10.
REQUEST(AAA,SN,*PF)
COPYBR(INPUT,AAA)
CATALOG,AAA,AUTOMAT,ID=INSTALL,XR=XYZ,PW=XYZ.
7/8/9
.PROC,AUTOMAT,DECK,TD=HY.OPTION1,OPTION2,...,OPTIONn.
IFE,STDS=SHYS.TRK7.
UPDATE,Q,D.8.*=-,I=DIREC7.

```

```

ELSE,TRK7.
UPDATE,Q,D,8,*==,I=DIREC9.
ENDIF,TRK7.
.DATA,DIREC7.
=ID JOBCARD
=D ACCOUNT.2
DECK,T0,MT1.
=DEFINE TD,OPTION1,OPTION2,....,OPTIONn.
=DEFINE (Put invariant defines here, such as ULIB)
=COMPILE DECK
.EOR
.DATA,DIREC9
=ID JOBCARD
=D ACCOUNT.2
DECK,T0,TD 1.
=DEFINE TD,OPTION1,OPTION2,....,OPTIONn.
=DEFINE (Put invariant defines here, such as ULIB)
=COMPILE DECK
6/7/8/9

```

### OBTAINING INSTALLATION JOBS

To process selected installation decks with appropriate options specified, run a job such as the following.

```

AUTO,T100.
ATTACH,OLDPL,DECKS,ID=INSTALL.
REQUEST,COMPILE,*Q.
ATTACH,AUTOMAT,ID=INSTALL.
AUTOMAT,deck,density,option1,option2,....,optionn.
ROUTE(COMPILE,DC=IN)
6/7/8/9

```

- deck        One of the decks available on the installation deck PL
- density    HY, HD, or PE tape density
- option<sub>i</sub>    Any appropriate option other than tape density as described in the DECK OPTIONS section

This technique works for all installation jobs except DST1 and DST2, which can be manually manipulated to add 844 0SY, 844 0SZ, 885 0SJ, and/or 66X 0MT controlware. In this case, selected jobs can be obtained in punch card form from the installation deck PL with a job similar to the following, or common deck CWARE can be modified to TRANSFER this controlware from a local file. The ATTACH of the local controlware file may be inserted at LOCALCC.1.

```

DECKS,T100.
ATTACH,OLDPL,DECKS,ID=INSTALL.
ATTACH,AUTOMAT,ID=INSTALL.
AUTOMAT,deck,density,option1,option2,....,optionn.
REWIND,COMPILE.
COPYBF,COMPILE,PUNCH.
6/7/8/9

```

- deck        One of the decks available on the installation deck PL
- density    HY, HD, or PE tape density
- option<sub>i</sub>    Any appropriate deck option described in the DECK OPTIONS section (other than tape density) that is not defined in the AUTOMAT procedure

This job can be used to punch any of the installation decks.

### DECK OPTIONS

Various options have been embedded in the job decks present on the installation deck PL under the control of =IF DEF directives for which values may be defined during extraction. The listing from deck REASON, described previously, shows the decks affected by defining the following options.

- BASE        The default installation (as opposed to EDITLIB or verification) type jobs are constructed to attach and update a corrective code file for use as input to UPDATE correction runs to produce upgraded product and system oldpls. Defining BASE during the extraction of installation jobs causes omission of the corrective code file steps. Refer to the appropriate software release bulletin for the products for which BASE should be defined.

HY, HD, and PE These options are used to select the appropriate REQUEST or LABEL statements in the installation deck according to the following.

HY 800 bpi 7-track  
HD 800 bpi 9-track  
PE 1600 cpi phase-encoded 9-track

One of these options must be selected.

**LIST** Places output from FORTRAN and SYMPL compilations and COMPASS assemblies on file LIST.

**0MT** Activates control statements and input file positioning to include the 66x (MTS) tape subsystem controlware in the file named CWARE from which all prefixed binaries are included. If 0MT is defined, the prefixed controlware deck is expected to be present in the appropriate input record of DST1 or DST2. Otherwise, it is assumed that a local file named CWARE, which contains this prefixed binary, has been attached or created at LOCALCC.1. The default residency for 66x and 67x tape drivers is central memory.

**0SJ** Causes the inclusion of 7155 controlware on the file named CWARE. If 0SJ is not defined, the local file CWARE created at LOCALCC.1 must include prefixed 0SJ controlware (refer to the 0MT description). If 0SJ is defined, the prefixed binary is expected to be present in the appropriate input record section of the installation deck.

**0SY** Causes the inclusion of 7054 controlware on the file named CWARE. If 0SY is not defined, the local file CWARE created at LOCALCC.1 must include prefixed 0SY controlware (refer to the 0MT description). If 0SY is defined, the prefixed binary is expected to be present in the appropriate input record section of the installation deck.

**0SZ** Causes the inclusion of 7154 controlware on the file named CWARE. If 0SZ is not defined, the local file CWARE created at LOCALCC.1 must include the prefixed 0SZ controlware (refer to the 0MT description). If 0SZ is defined, the prefixed binary is expected to be present in the appropriate input record section of the installation deck.

**ECS** Activates control statements to accommodate assembly of the CMR segments and creates library CMRLIB to hold the relocatable binary.

**OPTFTN** This symbol allows installation of the normal, 2-pass, optimizing FORTRAN Extended 4 compiler. If only TS mode is to be installed, OPTFTN should not be defined.

**TSFTN** This option permits installation of the time-sharing (TS) mode of the FORTRAN Extended 4 compiler. If it is not defined, FORTRAN Extended 4 is installed without the capability of TS mode. In addition to TSFTN, OPTFTN must be defined for proper installation of any product written in the FORTRAN Extended 4 language.

**NOTE**

If neither TSFTN nor OPTFTN is defined, the deck PL7I will not produce a compiler. When both TSFTN and OPTFTN are defined, as is applicable to the time-sharing compiler only, a compiler capable of handling both modes is installed.

**63CSET** Activates control statements to accommodate 63-character set installations and nominal execution field lengths required to accommodate conversion tables from a 64-character set to a 63-character set.

**DMGMNT** Activates assembly options to include the CDCS interface in COBOL 4 and COBOL 5; also affects nominal execution field length values.

Installation of QU 3 requires defining DMGMNT for DST1, DST2, and DST3.

To activate the CDCS 2 processing in COBOL 5, the value CD2 must be defined in conjunction with DMGMNT in PL60I and PL60I1.

**LOCLIB** Activates control statements in PL9I, PL9E, PL60I, PL60E, PL60O, PL66I, and PL66E to EDITLIB the binaries of the product into a local library instead of DST1 user libraries or the running system.

Defining LOCLIB in PL73I activates control statements to attach the COBOL 5 local library created by PL60I, PL60E, PL60O, or PL60I1 and execute this version of COBOL 5 during the installation of Data Catalogue 2.



NO1M1 Can be defined to avoid assembly of the low-speed multiplexer driver 1M1 in INTERCOM 4.

SRMS Can be defined in conjunction with setting installation parameter IP.SRMS to 1 in deck IPARAMS on PL1A and running job PL1HI prior to installing NOS/BE PL1B with SRMS defined. PL1BI accesses SRMSTXT during the assembly of 1PC-CVL.

0F3 Activates installation of the CCP 1 TTY binary.

0F4 Activates installation of the CCP 1 mode 4 binary.

0F7 Activates installation of the CCP 1 TTY and mode 4 binary.

CCP64 Activates statements necessary to build a 64K version of CCP 1 instead of the 48K default version.

NOCCP Activates selection of the 77K PASCAL compiler and 77K PASCAL cross-reference program.

CIOCP Activates inclusion of the 7000 connected I/O MUJ HELLO7. Use of this symbol requires that the updated PL1D (created by PL1DI) be assigned as input to PL12I or PL14I.

CATALOG Causes COBOL 4/5 conversion aid permanent files to be cataloged by job PL69I. Causes CDC CYBER cross system permanent files to be cataloged by job PL50I for subsequent installation of CCP 1 or CCI 3.

ULIB Invokes the use of the user library method for deadstart tape creation.

ES4IMS Activates control statements to produce IMS documentation for the enhanced station.

ES4XREF Creates a global cross-reference for the enhanced station SYMPL routines (MFSTAT and SPOT jobs). Defining this symbol requires that Sort/Merge, FORTRAN Extended 4, and FORTRAN Common Library 4 be present in the running system.

DIM Activates the capture of on-line maintenance software deadstart records.

CMU Activates inclusion of CMS test for compare/move unit hardware.

SAVE Reduces the number of tape assignments. All I decks that produce a tape that is used by another I deck catalog the necessary information as a permanent file. The receiving I job attaches this file instead of requesting the tape. Thus, all I jobs require one input tape and one output tape. In DST1, all user libraries are purged unless SAVE is defined. In DST2, the only tape assignment required is the output deadstart tape.

DL2 Activates selection of control statements to verify DDL 2 in PL55V. The default verification is DDL 3. This option also activates control statements in PL55I to select either PL56 or PL77 to extract the syntax table generator SYNGEN.

#### COMMON DECK MODIFICATIONS

ACCOUNT Common deck called by all job decks to allow insertion of a job statement and a valid ACCOUNT statement (=D ACCOUNT.2).

EXIT Allows use of the console display for abnormal termination of installation decks, and can be modified to provide different termination procedures. The following control statement sequence is suggested when reporting possible installation problems.

```

      =I EXIT.1
      REWIND(INPUT)
      COPYSBF.
      EXIT(S)
      REWIND(INPUT)
      COPYSBF.

```



LOCALCC  
 LOCALIN  
 READS

These three common decks allow local source code modifications. Every I suffixed deck contains calls to these three common decks. Additional control statements are inserted at LOCALCC.1, additional input sections (each section preceded with an =WEOR, 0 UPDATE directive) are inserted at LOCALIN.1, and any UPDATE \*READ directives are inserted at READS.1. The following examples illustrate the use of this facility.

Local modifications on an UPDATE oldpl.

```
=I LOCALCC.1
ATTACH(OLDPL,LOCPL,ID=LOCID)
UPDATE(Q,C=LOCPL1A)
RETURN(OLDPL)
=I LOCALIN.1
=/ NEED =WEOR, 0 TO MAINTAIN INPUT FILE POSITION
=WEOR,0
*C LOCPL1A
=I READS.1
*READ LOCPL1A
```

Local modifications in source format.

1. =I LOCALCC.1  
 ATTACH(LOCPL1A,ID=PL1A)  
 =I READS.1  
 \*READ LOCPL1A
2. =I LOCALCC.1  
 ATTACH (LOCCODE,LOCPL1A,ID=PL1A)  
 (The default file name LOCCODE is always read.)
3. =I READS.1  
 \*IDENT LOCPL1A

: Cards in extraction deck or procedure

SAVEPL

This common deck allows the installation, through the use of an =I SAVEPL.2 directive, to itemize or add additional operations to a new program library tape before it is returned. The newpl always has an lfn of PLxx at this point in the program.

Example: =I SAVEPL.2  
 ITEMIZE,PL1A,E,N.

#### DST1 AND DST2 COMMON DECKS

<u>Common Deck Name</u>	<u>Use</u>
CWARE (CWARE.2)	Common deck location containing a TRANSFER(*,CWARE) directive for inclusion of prefixed controlware on the new deadstart tape. Refer to OMT, OSJ, OSY, and OSZ under Deck Options.
CMRS (CMRS.1)	Empty common deck for inserting TRANSFER directives to include additional CMRs
XTRALIBS (XTRALIBS.1)	Empty common deck for adding additional system libraries. (A LIBRARY., REPLACE or ADD, and FINISH. span is required for DST1; an INCLUDE is required for DST2.)
MORENUC (MORENUC.1)	Empty common deck for adding additional NUCLEUS routines (with REPLACE directives)
MOREOV (MOREOV.1)	Empty common deck for adding additional overlays to SYSOVL (with REPLACE directives)
MORESYS (MORESYS.1)	Empty common deck to add more SYSLIB routines (with REPLACE directives)
MOREPP (MOREPP.1)	Empty common deck to add more PP routines (with REPLACE directives)
LOCALCC (LOCALCC.1)	Empty common deck for insertion of control statements to access additional files
SYSPROC (SYSPROC.2)	Empty procedure (SYSPROC) to which the user may add his own procedure control statements at =I SYSPROC.2. This procedure will be called after each deadstart.

## SPECIAL PURPOSE DECKS

### MINI

This deck is used to save the operating system and product set corrective code files from the batched corrective code (BCC) tape as permanent files. The following files are cataloged.

<u>BCC Tape File Number</u>	<u>File Name</u>	<u>Description</u>
1	OSMINIT	Operating system code
2	MINIT	Product set code

MINI must be run before all installation decks that require corrective code, since those decks attempt to attach OSMINIT or MINIT.

### ULIB

This deck must be run prior to the deadstart tape creation job (DST1) if the user library approach to installation has been selected (by including the =DF,ULIB directive when obtaining the installation jobs). This deck manipulates certain user library files in preparation for DST1. It combines files USERCC and USERNUC and creates a second cycle of files USERPP and USERPS. The second cycles of USERPP and USERPS are sequential files.

#### **NOTE**

A permanent file dump of the user libraries (ID=CCT) should be taken prior to running ULIB. If DST1 fails or if another product is to be installed, the permanent files can be reloaded and ULIB and DST1 can be run again.

### LIBS

This deck must be run first if the user library approach to installation has been selected. This deck catalogs all possible user libraries and inserts a dummy routine (ZZZ) in each. This dummy routine is later deleted from the libraries.

### DST1

This deck is a deadstart tape creation job which generates a system from user libraries. This deck is applicable only if the ULIB approach (=DF,ULIB directive included) has been selected for the other installation jobs.

### DST2

This deck has two purposes, depending on whether the running system method or the ULIB method of installation has been chosen. If the running system method of installation has been chosen, this deck creates a deadstart tape from the running system replacing the existing PL1A, PL1B, PL1E, and PL5 binaries with the updated versions. If the ULIB method of installation has been chosen, this deck copies the operating system binaries, created in a previous assembly, into the appropriate user libraries. This eliminates the necessity of reassembling the operating system in order to install subsequent products. No deadstart tape is created using the ULIB approach.

#### **NOTE**

Jobs DST1 and DST2 contain EDITLIB comment directives indicating alternative choices regarding central memory residency and INTERCOM driver choices. Users of these job decks are encouraged to review these comments before running the job decks.

### DST3

This deck creates a deadstart tape from the running system and CDC optionally replaces the deadstart diagnostic sequencer routines using existing PL5 binaries. CDC CYBER 171 sites with the no card reader configuration use the AUTO procedure to initiate this deck; when initiated, a CMR configured for a 2550 multiplexer is assembled and placed as CMR zero on the new deadstart tape. All other site configurations are discouraged from using this installation option.

### IPTEXT

This deck is a utility which replaces the running system text of the same name. Running IPTEXT ensures that references to any new or altered IPARAMS are accommodated during the assembly of the system and products (refer to 63CSET). Whenever any of the IPARAMS IP.C63, IP.PD, MODEL, OS.NAME, or OS.VER is to be changed from its default value when PL1AI is run, IPTEXT should be run. If IPTEXT is not run at this time, the altered values are not reflected in UPDATE and COMPASS.

### COMPCOM

This deck creates and catalogs the COMPCOM file. Because this file is produced in the PL2I job, this deck need only be used if PL2I is not run or if the file is destroyed. The file COMPCOM is used by the jobs that create the FORTRAN Extended 4 compiler (PL7) and the PL/I compiler (PL79).

### PLIAT

Job PLIAT is a utility deck used to equate SYSTEXT to CPCTEXT. None of the installation decks reference SYSTEXT; therefore, this job is not needed for the installation process. The default equivalence of SYSTEXT is IOTEXT as installed by CDC CYBER Record Manager installation decks and present on the unconfigured deadstart tape. This job may be run after the installation of CDC CYBER Record Manager and before the creation of a deadstart tape.

### SYSTEM TEXTS

Common decks included on the NOS/BE and CDC CYBER Record Manager program libraries are combined to form 15 system texts. The source location and contents of these common decks are as follows.

ACTCOM (PL1A program library)  
CPU program system action request macros

COMACIO (PL1A program library)  
CPU input/output macros

COMAFET (PL1A program library)  
File environment table generation macros

COMAREG (PL1A program library)  
Replacement for R= pseudo instruction

COMSHSP (PL1A program library)  
819 RMS definitions and macros

COMSRAS (PL1A program library)  
System communication symbols

CPSYS (PL1A program library)  
CPU input/output macros using CPC

IPARAMS (PL1A program library)  
NOS/BE 1 installation parameters

LMACOM (PL1B program library)  
CPU program loader request macros

PFCOM (PL1B program library)  
Permanent file macros

PPSYS (PL1A program library)  
PPU system definitions

SCHCOM (PL1A program library)  
Integrated scheduler macros

SISICOM (PL1A program library)  
Indexed sequential macros

STATCOM (PL1A program library)  
Enhanced station symbol definitions

CRMCOM (CDC CYBER Record Manager program library)  
CDC CYBER Record Manager user macros

Table I-1-2 shows the combination of these common decks into the various system texts required for full utilization of the product set. These texts are fixed in content except SYSTEXT; as released, SYSTEXT will contain ACTCOM, COMSRAS, and CRMCOM. At installation option, SYSTEXT may contain COMAFET, CPSYS, and SISICOM in lieu of CRMCOM.

TABLE I-1-2. COMMON DECKS AND SYSTEM TEXTS

System Text Name Common Deck Name	CMRTEXT	CPCTEXT	CPUTEXT	IOTEXT	IPTEXT	LDRTEXT	PFMTEXT	PPTEXT	SCHTEXT	SCPTTEXT	STATEXT	SYSTEXT DEFAULT	SYSTEXT ALTERNATE
ACTCOM		X	X	X						X		X	X
COMACIO			X										
COMAFET		X	X							X			X
COMAREG			X										
COMSHSP	X												
COMSRAS		X	X	X				X		X		X	X
CPSYS		X								X			X
IPARAMS	X				X								
LMACOM						X							
PFCOM							X						
PPSYS								X		X			
SCHCOM									X				
SISICOM		X											X
STATCOM											X		
CRMCOM				X								X	

The following texts are installed when CDC CYBER Record Manager, ALGOL 4, ALGOL 5, Sort/Merge, and FORTRAN Extended 4 are installed.

ALGTEXT This text is used only by ALGOL 4 programs.

ALG5TEXT This text is used by COMPASS coded routines with ALGOL 5 calling sequence.

SMTEXT COMPASS routines containing Sort/Merge macros require the specification of this text when assembled.

FTNMAC COMPASS routines generated by the FORTRAN Extended 4 compiler in E mode require the specification of this text when assembled.

The following text is cataloged when Shared RMS (PL1H) is installed.

SRMSTXT System routine MNT on PL1B requires the specification of this text when assembled if IP.SRMS is set to 1. The INTERCOM 5 routine MYQ on PL14 requires this text when assembled if Gemini is to be used on the system. The routine Gemini on PL1H also requires this text.

The system texts are constructed as a part of the installation process.

The following list shows the product name, the program library (PL) number, the section in which the product is discussed in this document, the BCC deck identifier, and the definable attribute causing the inclusion of the product in the DST jobs.

<u>Product</u>	<u>PL</u>	<u>Section</u>	<u>BCC Deck Identifier</u>	<u>DST Define †</u>
ALGOL 60 4	PL32	13	AGL,AGT	AGL
ALGOL 5	PL72	36	AL5	AL5
BASIC 3	PL57	21	BAS	BAS
CDC CYBER Control Language	PL70	26	CCL	-
CDC CYBER Cross System 1	PL50	15	XSY	-
CDC CYBER Database Control System 1	PL54	18	CDCS	CD1
CDC CYBER Database Control System 2	PL74	30	CD2	CD2
CDC CYBER Interactive Debug 1	PL82	32	ID1	ID1
CDC CYBER Loader 1	PL1E	16	LDR	-
CDC CYBER Record Manager 2 Advanced (Initial)	PL3B	4	AM1	AM1
CDC CYBER Record Manager 2 Advanced (Extended)	PL3C	4	AM2	AM2
CDC CYBER Record Manager 1 Basic	PL3A	3	SW1	-
COBOL 4/5 Conversion Aid	PL69	25	C45	-
COBOL 4	PL9	10	CL4	CL4
COBOL 5	PL60	24	CL5	CL5
Common Code Generator	PL83		CCG	
Communication Control INTERCOM 3	PL99	29	CC3,SCF	CCI2550
Communications Control Program 1	PL61	23	CC1	CC1
COMPASS 3	PL2	2	CPS	-
CTI (deadstart routines and test)	PL5B		CTI	-
Data Base Utilities 1	PL58	22	DBU	DU1
Data Catalogue 2	PL73	41	DC2	-
Data Description Language 2, SYNGEN	PL56	20	DL2	DL2
Data Description Language 3	PL77	31	DL3	DL3
Enhanced Station	PL1D†††	1	ES4	ES4
EXPORT High Speed	PL80	40	EHS	EHS
Factory Format Support (844-21 and 844-41)	PL1F	17	FMT	-
FORM 1	PL4B	6	FO4	FO4
FORTRAN Data Base Facility	PL66	35	FDB	FDB
FORTRAN Extended 4 (compiler)	PL7	9	FCC	††
FORTRAN Extended 4 (object library)	PL8	34	FCL	-
FORTRAN Extended 4 (time-sharing option)	PL7A	9	FCC	††
FORTRAN 5 (compiler)	PL63	38	FC5	FL5
FORTRAN 5 (object library)	PL64	39	FL5	FL5
FORTRAN 4/5 Conversion Aid	PL65	37	F45	F45
Gemini	PL1H	1	SC4H	-
INTERCOM 4	PL12	12	IN4	IN4
INTERCOM 5	PL14	28	IN5	IN5
Maintenance Package (including SYMPL)	PL6	8	CA4, SMP, SFT	CA4
NOS/BE (part A)	PL1A	1	OSA	-
NOS/BE (part B)	PL1B	1	OSB	-
On-Line Maintenance Software (CE diagnostics)	PL5A	7	DIM	DIM
PL/I 1	PL79	33	PLI	PLI
QUERY UPDATE 3	PL55	19	QU3	QU3
Shared RMS (844-21 and 844-41)	PL1H†††	1	SC4H	-
Sort/Merge 4	PL10	11	ST4	ST4
UPDATE 1, CDC CYBER Utilities 1, Common Memory Manager 1	PL1C	1	UPD	-
8-Bit Subroutines 1	PL4A	5	BE4	BE4

† A dash in this column means that the product is automatically included or is not applicable; a blank in the column means that the product is only cataloged on the system by the installation catalog job.

†† TSFTN governs the default field length of FTN. OPTFTN causes the inclusion of FTNMAC (refer to section II-9).

††† PL1D and PL1H comprise the multimainframe module 1 package.

102684192



**PART II**



## HARDWARE CONFIGURATION

The target minimum hardware configuration consists of the following.

- 1 CDC CYBER 170 Model 171, 172, 173, 174, 175, 176, 720, 730, 750, or 760; CDC CYBER 70 Model 71, 72, 73, or 74; or 6000 Series computer with a minimum of 65K central memory and ten peripheral processor units
- 1 844 disk subsystem (844-21 or 844-41) or one 885 disk subsystem
- 1 512 or 580 printer, or line printer on a 734-1/CDC CYBER 18 terminal that is physically located with the central computer and is driven by a 255x communication subsystem
- 1 405 card reader, or card reader on a 734-1/CDC CYBER 18 terminal that is physically located with the central computer and is driven by a 255x communication subsystem. (The terminal card reader can be used only for source deck submission and not for binary decks.)
- 2 Magnetic tapes from either 667, 669, 677, or 679
- 1 255x communication subsystem with at least 65K memory (required if the site does not have a 405 card reader or if the site requires INTERCOM 5)

## RELEASE MATERIALS

Materials in the NOS/BE 1 release package consist of the following.

- PL1 A, PL1 B NOS/BE program libraries
- PL1 C UPDATE release tape, including COPYL, ITEMIZE, and CMM
- PL1 D station release tape†
- PL1 E CDC CYBER Loader 1 release tape
- PL1 F Factory Format Support (844-21 and 844-41)
- PL1 H shared RMS release tape†
- PL2 COMPASS 3 release tape
- PL3 A CDC CYBER Record Manager Basic Access Method (BAM) 1 release tape
- PL3 B CDC CYBER Record Manager Advanced Access Method (AAM) 2 release tape
- PL3 C CDC CYBER Record Manager Advanced Access Method (AAM) 2 release tape
- PL4 8-Bit Subroutines 1 and FORM 1 release tape
- PL5 On-Line Maintenance Software release tape
- PL70 CDC CYBER Control Language release tape
- Unconfigured deadstart tape
- Installation deck program library
- Small binary coldstart card deck

For model 176 installation, the following is also included in the operating system release package.

- PL1 G 819 RMS

PL1 G is a three-file tape, structured as follows.

- File 1 Program library
- File 2 Binary of the 819 FLPP driver
- File 3 Binary of CMR segments to support 819 RMS

The NOS/BE program library tapes (PL1A through PL1H) contain the source programs for all routines comprising NOS/BE 1. PL1A and PL1B contain one file each; assembled binary is not included on these tapes. PL1C contains a program library as file 1, the absolute binaries of UPDATE, COPYL, and ITEMIZE as file 2, and the relocatable binaries of CMM as file 3. PL1E, PL1F, and PL1H contain a program library as file 1 and assembled binary as file 2. PL1D is a six-file tape structured as follows: file 1 is a program library, and files 2 through 6 contain PPU absolute binaries, station absolute binaries, spun off task absolute binaries, station relocatable binaries, and spun off task relocatable binaries, respectively.

† Separately licensed products, although PL1D and PL1H together comprise the Multimainframe Module 1 package.

The unconfigured deadstart tape contains only the products NOS/BE (including UPDATE, CMM, and CDC CYBER Loader, but excluding the station and shared RMS), COMPASS, Factory Format Support, On-Line Maintenance Software, CDC CYBER Record Manager Basic Access Method, SYMPL, and CDC CYBER Control Language. The unconfigured deadstart tape corresponds to the release program libraries. Six CMRs are present on this tape. The first reflects a blank EST, and the second and third reflect released configurations without CYBER 170 model 176 capabilities; the fourth reflects a blank EST, and the fifth and sixth reflect released configurations with model 176 capabilities. The first and fourth CMRs allow keyboard entries of up to eight tape drives and six RMS equipments. While the released CMRs are generally usable for channel and equipment numbers, these CMRs allow up to 8 tape drives and 12 RMS equipments. Refer to the Software Release Bulletin for applicable controlware part numbers for 0SY, 0SZ, 0SJ, and 0MT, which are a part of the unconfigured deadstart tape deadstart records.

The small binary coldstart card deck is for 66x coldstart-type deadstarts. If it is used, the 66x unprefixed controlware deck must follow. (A tape coldstart using the 7152 Mass Storage/Magnetic Tape Controller is described later in this section.)

Content, structure, and use of the COMPASS, CDC CYBER Record Manager, FORM, and CDC CYBER Control Language release tapes are discussed in the section devoted to each of these products.

Required supplements to this package include the following.

- PL6 SYMPL 1 Maintenance Tools program library
- PL7 FORTRAN Extended 4 Compiler program library
- PL8 FORTRAN Extended 4 Object Routines program library

Provided as a required supplement to the basic release package, PL6 (SYMPL) is needed for use in installing the NOS/BE system (PL1B), the enhanced station, FORM, ALGOL, SYMPL, BAM, AAM, CDCS 1, DDL, DBU, COBOL 5, BASIC, CDCS 2, and QU 3. PL7 and PL8 are necessary for complete installation of PL1B and numerous other products in the total product set.

#### NOTES AND CAUTIONS

The Central Exchange Jump switch must be disabled when using the release deadstart tape.

Because PP monitor was restructured at NOS/BE 1.4, all PP programs written by installations must be reassembled using PPTXT from the NOS/BE 1.4 release. Use of PP binaries from earlier systems causes unpredictable results, including system crashes.

All relocatable binary programs that issue dayfile and/or B-display messages written for systems prior to NOS/BE 1.1 must be regenerated under NOS/BE 1.4 via assembly or compilation because of features added to MESSAGE processing for NOS/BE 1.1 (PSR summary level 439/439).

NOS/BE is released with all software assembled to support integer multiply. To ensure the proper execution of all code related to this hardware capability, the following change orders must be installed.

<u>Computer Types and Models</u>	<u>Change Order Number</u>
6600 Models A, B, C	CA26938, CA30886
6600 Models D, E	CA26379, CA31029
6200/6400/6500 All models	CA26792, CA30638, CA28539
6700 Models A, B, C	CA27065, CA30966
CDC CYBER 70 Models 72 and 73	CA30639
CDC CYBER 70 Model 74	CA31029

In addition, the following FCOs listed should be installed as they become available. These FCOs will make underflow results obtained from normalized numbers with zero exponents positive to ensure consistency across the product lines. The lack of these FCOs will not have a negative impact on the operating system or product set members; however, the CE diagnostic routine, CT3, may fail if a certain set of random operands are generated for the multiply unit. For additional information concerning CT3 and integer multiply, consult the discussion of installation parameter INTMULT in the NOS/BE On-Line Maintenance Software Reference Manual.

<u>Computer Types and Models</u>	<u>Change Order Number</u>
6200/6400/6500 All models	CA33439
6700 Models A, B, C	CA33439
6700 Model D	CA32988
CDC CYBER 70 Models 72 and 73	CA32988

The deadstart tape and the system device must be on different channels. This is necessary because deadstart keeps two data streams going at one time.

When NOS/BE is run on a 6500 or 6700 using IP.XJ=1 or 2, FCO CA23065 must be installed to prevent both CPUs from being in monitor mode simultaneously.

ECS I/O buffering and swapping of non-MUJ jobs to ECS cannot be activated if 819 RMS is installed. L.ECSSWP defaults to 20B when 819 RMS is installed to allow MUJ jobs to swap to ECS.

ECS I/O buffering is not supported for private device sets.

If the OUTPUT file is rewound but no other action is performed on the file, the OUTPUT file will be evicted; a skip to EOI is not performed prior to writing the job dayfile on OUTPUT.

The values BASE and IRADR as described later under Deadstart Installation Parameters must be observed.

Support of the 6603, 6638, 821 and 854 devices is not provided.

Support of the 7611-11 Station is not provided.

Device overflow is not allowed between device sets.

A private device set may contain 841 devices, 844-2x devices, 844-4x devices, or 885 devices, but each private set may contain only one type of device.

FCO CA35742 must be installed to avoid a CPU A hang with monitor abort on a CDC CYBER 74-28 which will otherwise occur when a user job executing in CPU A under control of MODE,0. executes an instruction which attempts an out-of-range address.

FCO CA36100 must be installed on the 844 Buffer Controller (FA710 or FA719) to use the A08 controlware.

Code is activated to utilize FCO CA37722 (PP halt on CM read error), applicable to CDC CYBER 170/Model 176 systems and any other model of the CDC CYBER 170 series at production level C or D. Although this FCO is not essential to system operation, it improves system reliability and should, therefore, be installed.

#### NOTES AND CAUTIONS (SHARED RMS ENVIRONMENT)

The available RB count displayed in the DSD V display for shared devices is not assured to be accurate at all times.

A public device set with the system set attribute cannot be shared.

FCO CA35682, FCO CA35683 and the latest release of the 844 controlware are required for shared RMS.

If RMS deadstart is used, it is recommended that neither the RMS deadstart device nor any controller accessing the device be accessible from another mainframe.

#### NOTES AND CAUTIONS (TAPE SCHEDULING)

The installation parameter IP.SCHDE determines whether 9-track units are scheduled by device or by density.

Setting IP.SCHDE to a nonzero value enables automatic scheduling by density. The job statement parameters for 9-track tape resources can be Ntk, HDk, PEk, or GEk, where k is the number of tape units at each density to schedule for the job, and NT is equated to the installation default density as defined by parameter IP.NDEN in deck CIOCOM. When IP.SCHDE is nonzero, jobs containing requests for 9-track tapes with the density specified as other than the default density must include the corresponding density specifications on the job statement.

Setting IP.SCHDE to zero disables automatic scheduling by density. Tape job statement parameters NT, HD, PE, or GE are allowed but are not required. Nine-track tape units scheduled as HD, PE, or GE are added to the count of units scheduled as NT.

The IP.SCHDE=0 option will be removed in a future NOS/BE release, at which time the density specification will always be required on the job statement. Therefore, setting IP.SCHDE to zero at this time allows installations to give their users time to convert job statements to include density parameters before the requirement for them becomes effective.

## NOTES AND CAUTIONS (ATS 679 TAPE DRIVES)

If tape resource scheduling by density is not enabled (IP.SCHDE=0), deadlocks (which may require jobs to be rerun or dropped) may occur because 679/GCR (group coded recording) tape drives do not have 800 epi recording capacity, but are considered available for assignment to any 9-track tape request. Deadlock situations can be avoided by one of the following operational procedures.

- The parameter IP.SCHDE can be set to a nonzero value. Refer to Notes and Cautions (Tape Scheduling).
- Tape previewing can be used to manually schedule jobs requiring 679 tape drives. This requires that the installation adopt a vsn convention for 679 tape reels, enabling the operator to recognize requests for such tapes on the P display. Jobs can be manually initiated after all tape resources have been made available. The limitation of this method is that all tape requests are not shown on the previewing display in the case of multimainframe systems, macro requests, or CCL procedure files.
- 679 scheduling deadlocks can also be avoided by ensuring that all 679 tape drives remain logically OFF in the equipment status table. In response to a request for a 679 tape, the operator must turn the tape drive logically ON, assign it to the requesting job (or allow automatic assignment if labeled), and turn the drive logically OFF. Tape drive overcommitment scheduling cannot be used with this method (bit S.OCJI of IP.TSG must be set to 0).

Use of 6250 cpi density on 679-7 tape drives is supported only on a CDC CYBER 170 mainframe with 2XPPU speed enabled (IP.PPS2X set to 2). Use of 6250 cpi density in an unsupported configuration may result in lost data during deadstart.

Tape reels used for recording at 6250 cpi must be certified at 3200 fci or greater.

## INSTALLATION PROCEDURES

Installation of NOS/BE requires customizing to conform to the site's hardware and software specifications by selection of the following.

- General installation parameters within IPARAMS
- Tape processing installation parameters within CIOCOM
- CMR configuration parameters
- Deadstart installation parameters
- Permanent file installation parameters
- Scheduling parameters
- Loader installation parameters
- ECS installation parameters
- UPDATE installation parameters
- Symmetric/Replacement Station installation parameters
- Gemini load leveling installation parameters

Once parameters have been selected, configured program libraries and a deadstart tape can be created by running the model jobs.

## GENERAL INSTALLATION PARAMETERS (IPARAMS)

General installation parameters related to NOS/BE are defined within the COMDECK IPARAMS. IPARAMS is listed in the routines IPTXT and CMR. Other installation parameters are described elsewhere in this and other sections of this document.

Assigned (default) values and descriptions follow. The parenthetical value is the default value as set on the released program library.

The default values of the IPARAMS configuration parameters are defined with the CEQU or CMICRO macros, so that an installation can insert all modifications at one given place. The CEQU and CMICRO macros are used to define variables conditionally. Since they are effective only if the variables have not been previously defined, any modifications must precede the default definitions.

Symbols can be defined by EQU or CEQU except for OS.ID and MODEL, which are micros and must be defined by CMICRO.

Examples of changes for deck IPARAMS

```
*I IPARAMS.15
IP.LINK CEQU 0
MODEL CMICRO 3,(175)
IP.CSET EQU IP.C64.2
```

The following list constitutes the extent of installation changeable symbols in IPARAMS. Changes to the default values listed following should be made at IPARAMS.15 in an update of PLI A. The IPARAMS common deck also contains symbols IP.ILCMD, IP.IUSID, IP.1M1, IP.1WB, and IP.1ZZ. These symbols are described in the INTERCOM 4 section.

HF.LIST (P74,S7)

Micro whose value specifies the presence of certain hardware features in the configuration on which the product set is being installed. HF.LIST should always be supplied in addition to the MODEL micro, since use of various hardware features by the product set is conditional on HF.LIST. However, if HF.LIST is not defined, a default value that is based on the MODEL micro and assumes no optional hardware is used. The default HF.LIST based on MODEL is a temporary capability that will be removed in a future release. The following entries can be defined in HF.LIST.

- C Compare/move unit (CMU) hardware is present.
- L Large central memory (LCM) is present. This is memory for which direct access instructions (014 and 015) are defined. It exists on CYBER 170 model 176 mainframes.
- 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, this entry should be omitted. Following are instruction stack loop sizes for the given mainframes.

<u>Seven Words</u>	<u>Ten Words</u>
6600	CYBER 170 model 175
CYBER 70 model 74	CYBER 170 model 176
	CYBER 170 model 750
	CYBER 170 model 760

Px Type of central processor; x can be one of the following values.

- S Serial type CPU. S should be used for 6200, 6400, 6500, CYBER 70 models 71, 72, and 73, and CYBER 170 models 171, 172, 173, 174, 720, and 730.
- 74 6600, 6700, and CYBER 70 model 74.
- 175 CYBER 170 models 175, 750, and 760.
- 176 CYBER 170 model 176.

The processor type defaults to PS if HF.LIST is defined but the processor type is omitted.

Default values for HF.LIST when HF.LIST is not defined are as follows:

<u>MODEL Micro Value</u>	<u>HF.LIST Default String</u>
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
Any other	PS

Duplicate parameter entries (such as two Px entries) are not allowed.

A central processor type of PS, P74, or P175 can be used when defining HF.LIST for a product set intended to be run on multiple mainframes. Stack size can be included (even if not all the mainframes have a stack), but C and L must not be included unless the respective features exist on all the mainframes in the configuration. The resulting product set will not necessarily perform optimally on any of the mainframes, but will perform better on a parallel processor (such as a 175) if that processor type is set in HF.LIST.

IP.CMU (0)

If nonzero, Compare/Move Unit hardware is present. If set to 1, the system will not run on a non-CMU mainframe (such as a model 175 or a 6600).

IP.ACNT (0)

If zero, normal control statement processing occurs. If set to 1, the job statement is copied to RA+70B through RA+77B; the CPU program ACCOUNT is then loaded and executed. If set to greater than 1, the first statement following the job statement is copied to RA+70B through RA+77B before the CPU program ACCOUNT is loaded and executed. No dayfile message is issued when ACCOUNT is called.

If IP.ACNT=1 and IP.ARCH=1, the instructions in the following three paragraphs must be followed.

If an installation has system modifications which require accounting information before a job can come to a control point, changes must be made to routine 1PF to insert appropriate information into the control stream of the job that performs archive file retrieval.

Any accounting information needed by the installation must be inserted into the control statement buffer. These statements, including the job statement, must be formatted according to installation procedures, using DIS or DATA statements. Each DIS or DATA that completes a card must be followed by a call to the PAD macro, which pads the card with zeros.

JOBNAME must be set equal to a valid five-character, local file name to be used in setting up the input FNT for the archive retrieval job. 1PF adds two random digits to this jobname before storing it into the FNT. JOBNAME should be the same as that used on the job statement. No CM, tape, or priority requirements need be on the job statement, as 1PF sets up the input FNT with all such requirements satisfied.

Example 1 (accounting information on job statement)

```
*D FEAT61B.1 NEAR 1PF.180
CARD1 DATA H*JBNME,CMTIME,*
*D FEAT61B.2 NEAR 1PF.180
TAPES DIS ,*MT1. ACCOUNTING INFORMATION *
```

Example 2 (accounting information not on job statement)

```
*D FEAT61B.1 NEAR 1PF.180
CARD1 DATA H*JBNME,CMTIME,*
*D FEAT61B.2 NEAR 1PF.180
TAPES DIS ,*MT1,*
PAD
CARD2 DIS ,*ACCOUNT(X,Y)*

*D 1PF.206
JOBNAME DIS ,*FGHIJ*
```

IP.CP (6)

If set to 6, default punch mode is 026. If set to 9, default punch mode is 029. The alternate punch mode is selectable by ROUTE parameters.

IP.ARCH (1)

Archive feature; permanent files dumped under a mode 2 permanent file DUMPF

- 0 No longer will have a PFC entry and will not be retrieved from tape at ATTACH time.
- 1 PFC entries are retained and are retrieved from tape at ATTACH time (refer to IP.ACNT).
- 2 PFC entries are retained but are not retrieved from tape at ATTACH time.



IP.CPLM (5)

Installation-defined number of central processor (CP) seconds by which a job is incremented if the CP time limit (specified on the job statement or IP.STL) is exceeded or if the job requires EXIT or error processing.

IP.CPLM also specifies the number of additional CP seconds that an INTERCOM user can have after the session time limit (defined in the password file) is reached.

IP.DBAL (3777B)

Installation-defined value for default batch access level. This value should be set to  $2^{IP.IACES-1}$ . The system assigns this value to any job that is not interactive, and the loader checks the value before loading system library resident programs. (The value for interactive jobs depends on the access level assigned in the password file.)

IP.CR (69D)

If set to 6, all BCD cards are read as if punched by a 026. If set to 9, all BCD cards are read as if punched by a 029. If set to 69, all BCD cards are read as if punched by a 026; however, if a job card or a 7/8/9 card has 29 punched in columns 79-80, all following BCD cards in that job are read as if punched by a 029, until a following 7/8/9 card changes the mode again. If set to 96, the inverse is true: 029 is default and job and 7/8/9 cards may switch to 026. The card reader routine, 2RC, treats all level 17 7/8/9 cards as end-of-file for compatibility with JANUS.

IP.CSET (IP.C64.1)

Using the IPARAMS symbols indicated, an installation can select one of two graphic character sets, CDC Scientific or ASCII. It can also independently select one of two character set sizes, 63 or 64 characters. The default character set is the CDC 64-character set. The IPARAMS modifications used to select each of the other three possible character sets are as follows.

ASCII 64-character set	IP.CSET	EQU	IP.C64.2
CDC 63-character set	IP.C63	EQU	IP.C64.1
	IP.CSET	EQU	IP.C63
ASCII 63-character set	IP.C63	EQU	IP.C64.2
	IP.CSET	EQU	IP.C63

The relationship chosen for IP.CSET, IP.C63, IP.C64.1 and IP.C64.2 must hold constant when all products referencing them are assembled for inclusion in a deadstart tape.

The character sets are described in detail in the NOS/BE 1 Reference Manual.

IP.C176 (0)

If nonzero, code to support CDC CYBER 170 Model 176 systems is assembled. This option is automatically enabled if IP.819 is equated to one.

IP.DSRMS (1)

If nonzero, the code to perform a deadstart from an RMS device is assembled and replaces the code to read 60x/65x tapes for deadstart. If zero, the 60x/65x code is assembled and the RMS code is not. As a result, 60x/65x tapes cannot be used for deadstart when RMS deadstart is in effect. If the site must use 60x/65x tapes for deadstart but also wants to use the RMS deadstart feature, a minimal recoding effort is required to replace the 66x/67x tape code with code for 60x/65x tapes. The default value for IP.DSRMS is nonzero; however, the code on the unconfigured deadstart tape was assembled with IP.DSRMS set to zero.

IP.ECSB (0)

If zero, the ECS extension code is not assembled. If nonzero, the ECS extension code is assembled and the ECS installation parameters are activated. This parameter must be nonzero for ECS systems.

IP.IACES (11D)

Installation-defined size of the access level field. It should be identical to the INTERCOM definition of IP.IACES (refer to section 12). It is defined here for 1AJ and the CDC CYBER Loader.

IP.IOLM (100B)

Installation-defined number of I/O seconds by which a job is incremented if it exceeds I/O time limit, as specified on the job statement or IP.SIOL, if needed for EXIT or error processing.



IP.IQD (6)

Input queue priority increment delay. The input queue priority is incremented by one every  $2^{**IP.IQD}$  seconds (0-11).

IP.IQPW (3)

Input queue priority weight (0 through 12). When a job is being considered for initiation, its effective input queue priority is  $P * 2^{**n - IP.IQPW} + A$  where P is the job statement priority, A is the age factor, and n satisfies the relation  $4000g \leq IP.LVF * 2^{**n} \leq 7777g$ .

The aging process does not allow the age factor to exceed a maximum value of  $2^n - 1$ . Thus, if IP.IQPW=0, a job with a higher job statement priority will always be initiated before a job with a lower job statement priority, regardless of the length of time the lower priority job has been waiting in the input queue. If IP.IQPW=12, job initiation is determined solely by age factor; job statement priority will not affect the choice. Selecting a value between these extremes allows both factors to be taken into consideration and provides a means for weighting one factor over the other.

IP.LINK (1)

Maximum number of links connected to this mainframe. If IP.LINK = 0, the linked 6000/7000 and the linked 6000/6000 command/display code will not be assembled. The following DSD overlays will not be assembled: 8YA, 8YB, 8YC, 8YD, 8YE, 8YF, 8YG, 8YH, 8YI, 8EB, 8EC, 8EE, 8EH, 8EJ, 8EP, 8YR, 8YU, 8YZ.

IP.LVF (70B)

Lowest fixed priority (2 through 7777g). A fixed priority does not age and cannot be specified by a user (IP.LVF must be greater than IP.MPR). The value of IP.LVF also affects the processing of input queue priorities (refer to IP.IQPW), output queue priorities (refer to IP.OPRI), and job queue priorities. The calculation of a job queue priority includes a weighting factor as follows:

$$P * 2^{**n - 6} * 10g$$

where P is the job statement priority and n satisfies the relation  $4000g \leq IP.LVF * 2^{**n} \leq 7777g$ .

IP.MCPU (1)

Installation option to define maximum number of CPUs to be used by system. The value 1 produces the most efficient code for use on a single CPU. The system runs on a dual CPU machine, but uses only one CPU. The value 2 produces a variant of MTR which runs on a dual CPU machine using both CPUs or less efficiently on a single CPU machine.

IP.MECS (0)

Maximum number 0 to 7777g of 1000g word blocks of ECS direct access that may be assigned in response to a job statement EC parameter, RFL statement, or MEMORY macro. This value determines whether sections of code are to be assembled within the system to handle ECS allocation. This parameter must be nonzero for ECS systems. IP.MECS should not be set equal to direct access total length as about 20K octal is used for ECS system segments area.

IP.MFL (14000B)

Maximum amount of central memory field length that may be assigned to a user job. A user cannot request more than IP.MFL field length on a job statement or with MEM or RFL. For additional information, refer to Scheduling Parameters. IP.MFL must not exceed 377700g.

IP.MMS (100B)

Maximum mass storage limit, 1 through 7777g, that may be specified by PRUs/100 (octal) on a LIMIT statement.

IP.MPPU (10D)

The maximum number (7 through 20) of peripheral processors in the configuration of any of the CMRs on the deadstart tape. A value of 20 allows execution on a 10-PPU machine at the cost of reduced central memory availability. The value 10 does not allow execution on a 20-PPU machine.

IP.MPR (20B)

Maximum priority (1 through IP.LVF-1) a user can specify on his job statement. If a user specifies a higher priority, the default IP.SPR is used.

IP.MSCT (0)

Maximum decimal number of messages (1 to 4095) that may be entered into the dayfile by a single job. Only messages sent through MSG are counted. If zero, no maximum will be considered. Setting IP.MSCT  $\neq$  0 may cause some installation jobs to fail because of excess dayfile messages.

IP.MSLM (200B)

Installation-defined number of mass storage PRUs by which a job is incremented if it exceeds mass storage limit, as specified on the LIMIT statement, if needed for EXIT or error processing.

IP.MTL (77777B)

Maximum CP time limit in seconds, 0 to 77777g, that may be assigned to a job. Both 0 and 77777 are considered infinite.

IP.NDFS (1)

Number of dayfile copies on output. Up to 4095 may be specified.

IP.NJFL (20B)

FL/100B assigned to batch jobs when first assigned to a control point. Range (1 to IP.MFL). The default value allows execution of job setup utilities.

IP.OPRI (0)

Specifies whether the size of a file affects its output queue priority. If IP.OPRI is zero, the priority is  $P \cdot 2^{n-6} \cdot 100g + 1$  where P is the job statement priority and n satisfies the relation  $4000g \leq IP.LVF \cdot 2^n \leq 77777g$ . If IP.OPRI is nonzero, the output queue priority is  $P \cdot 2^{n-6} \cdot 100g + 2^{n-1} \cdot S \cdot 2^{n-10}$  where P is the job statement priority, S is the file size in PRUs, and n is as previous described. If the file size exceeds 1777g PRUs, the IP.OPRI=0 is used.

IP.OQD (10B)

Determines period for incrementing priority of a job in the output queue. This period is  $2 \cdot IP.OQD$  seconds. Legal values for IP.OQD are 0 through 13g.

IP.PD (6)

Default print density in lines per inch. This parameter is used only by products which support it in their documentation. Legal values are 6 or 8.

IP.PFRP (5)

Default retention period in days for permanent files cataloged without explicitly defined retention periods. The range of values is 0 through 999.

IP.POSFL (5)

Field length/100g reserved for use by ISO for requesting positive field length. Positive field length is not available to user jobs and can be considered part of CRM. Positive field length is allocated internal to the system for swapout use only. Range (4 to 10g).

IP.PPS2X (2)

All peripheral processors' major cycle time (CDC CYBER 170 only)

1= 1000 nanoseconds

2= 500 nanoseconds

IP.PS (60D)

Default page size in lines per page. This parameter is used only by products which support it in their documentation.

IP.RM (IP.HT)

Default recording mode of an 844 disk pack.

IP.HT Half-track recording mode

IP.FT Full-track recording mode

**NOTE**

Full track recording mode may be used only on an 844 accessed by at least one 7154 or 7155 controller in a CDC CYBER 170 system with the 2XPPU speed in effect (IP.PPS2X set to 2).

IP.SCHDE (0)

Tape scheduling for 9-track units

- 0 Disables tape resource scheduling by density.
- 1 Enables tape resource scheduling by density. Job statement processing and all 9-track tape unit scheduling are based on user density requests. Request and label statements must match job statement density requests.



IP.SECS (0)

Default number of direct access ECS blocks (1000 octal words) to be assigned to a job if not declared on job statement; range zero to IP.MECS.

IP.SFL (50000B)

Default central memory field length (octal) to be assigned to a job if not declared; range 100 to IP.MFL.

IP.SIDLE (1)

If nonzero, code to support IDLE mode is enabled. This code should be enabled if the system checkpoint capability or full Status/Control register monitoring is desired.

IP.SIOL (0)

Default I/O time limit in octal seconds (0-77777B) to be assigned to a job if not declared on the job statement. A value of zero is considered infinite. If IP.SIOL is set to a value other than zero, TDS should be assembled with an I/O time limit of zero on the job statement for the EDITLIB (SYSTEM, RESTORE) job at EDITPRUF and the LDCMR job at LDCMPRUF. The I/O time limit on the archived retrieval job should also be set to zero (near 1 PF.180).

IP.SMS (0)

If nonzero, the default mass storage PRU limit a job can use, divided by 100 (octal). All jobs, therefore, proceed as if a LIMIT statement with value IP.SMS were in the job deck. Refer to the LIMIT statement in the NOS/BE 1 Reference Manual. ISI assembles with a type 7 error (refer to the COMPASS Version 3 Reference Manual, Error directory) when IP.SMS exceeds 7777B, but the resulting code is correct. IP.SMS must not exceed 377777B (17 bits).

IP.SPR (10B)

Default priority given to a job if no priority specified on job statement. Range 1 to IP.MPR.

IP.SPT (0)

If zero, no Scheduler performance execution statistics are returned. If nonzero, such statistics are returned.

IP.SRMS (0)

If nonzero, first mainframe to deadstart appears on the operator option matrix during deadstart. Also, certain key code for RMS sharing is controlled with this parameter; thus, if shared RMS (PLI H) is not installed, IP.SRMS should be zero.

IP.STL (100B)

Default central processor time limit in octal seconds (00 to IP.MTL) to be assigned to a job if not declared on the job statement. Values of zero and 77777B are considered infinite.

IP.TCPUB (4)

CPU time for CPU A is accumulated at a rate that is IP.TCPUB/4 times greater than actual time used. The intention of this parameter is an attempt to equalize the time that will be accrued on either CPU of a 6700 or CDC CYBER 74-2x. It could also be used on a single CPU. An installation that has both a model 73 and a model 74 could use IP.TCPUB 8 on the model 74 to equalize the effect of the time limit on either machine.

IP.TYPE (6600)

Determines the type of central processor to be used by the operating system [6600 (CYBER 170 Model 175, 176, 750, 760, or CYBER 70 Model 74), 6400 (CYBER 170 Model 172, 173, 174, 720, 730, or CYBER 70 Model 71, 72 or 73) or 6500] for generation of optimal code. Acceptable values are 6400 and 6600.

IP.UP (10B)

Determines the permissions granted to a user who has specified the installation-defined universal password. This is a 4-bit field in which each nonzero bit signifies the type of permission granted.

<u>Bit Position</u>	<u>Permission Granted</u>
3	Control
2	Modify
1	Extend
0	Read

IP.US (0)

When the installation reserves space (user slot) in the Permanent File Catalog for information to be saved with each permanent file, this parameter is the space length in central memory words. 20D is the maximum value supported.

IP.XJ (-1)

Values, which pertain to routine CMR only, include the following.

- 0 Computer does not have the central exchange jump feature; central monitor is simulated.
- 1 Make use of central exchange jump feature.
- 2 Make use of central exchange jump feature including the MAN instruction.
- 1 Central exchange jump feature is not to be used. An exchange jump protection program is included to protect the system against an accidental execution of an exchange jump instruction.

When multiple CMRs are assembled, any of which are assembled with a value of 0 or -1, MTR must be assembled with a 0 or -1 value.

IP.819 (0)

If nonzero, code to support the 819 RMS subsystem is activated. LDCMR errors occur if this code is activated without installing code from PL1G.

IP.YMD (MDY)

Micro which shows format of date to be typed in at deadstart. The six possible permutations of the letters MDY constitute the range of this parameter.

MODEL (74)

Micro, used by the product set members for optimal code generation, whose value is the CDC CYBER 70 or CDC CYBER 170 model number corresponding to the type of central processor for which code is to be generated and optimized. Acceptable values are 71, 72, 73, 74, 171, 172, 173, 174, 175, 176, 720, 730, 750, or 760. The recommended value for a 6400 or 6500 is 73, and the recommended value for a 6600 or 6700 is 74.

**NOTE**

For use on a CYBER 170 model 176 without LCM, FORTRAN Extended 4, FORTRAN 5, FCL 4, FCL 5, and the Common Code Generator portion of PL/1 must be prepared with a MODEL micro setting of 175; all other product members can be prepared with a setting of 175 or 176.

OS.ID (NOS/BE 1.4)

System identification micro used by the product set members for displaying the operating system name and version number in generated program binaries.

TAPE PROCESSING INSTALLATION PARAMETERS (CIOCOM)

The default values of the CIOCOM configuration parameters are defined with the CEQU or CMICRO macros, so that an installation can insert all modifications at one given place. The CEQU and CMICRO macros are used to define variables conditionally. As they are effective only if the variables have not been previously defined, any modifications should precede them.

Installation parameters specifically oriented to tape processing are defined within the COMDECK CIOCOM. CIOCOM is listed in the routine CMR. Assigned (default) values, other tested values, and descriptions are as follows. Changes to default values should be made at CIOCOM.6 in an update of PL1A.

IP.CBKSP (1)

If one, controlled backspace is available in all controllers for 65x drives; if zero, it is not installed.

IP.D7DN (IP.TDEN)

Density for ATS/MTS deadstart dump tapes on 7-track units

2	800 bpi
other	556 bpi



IP.D9DN (IP.NDEN)

Density for ATS/MTS deadstart dump tapes on 9-track units

2	800 bpi on MTS and non-GCR ATS, 6250 cpi on GCR ATS.
other	1600 cpi

IP.NBCD (0)

9-track default conversion mode (0=ANSI, 1=EBCDIC)

IP.NBRK (0)

If zero, system noise records are used in write recovery at densities other than 1600 bpi. If one, they are not used. It is recommended that an installation run with noise bracketing enabled to take advantage of increased reliability on tapes which are not destined for interchange, and that users who are creating tapes for interchange purposes include the IB parameter on their tape requests.

Study has shown that the use of noise brackets on phase encoded tapes has not increased their reliability. For this reason, noise brackets are never written on phase encoded tapes.

IP.NDEN (3)

Density for label and data on 1/2-inch 9-track tape, if not declared on REQUEST or LABEL statement. (1=6250 cpi, 2=800 bpi, 3=1600 bpi)

IP.NTCN (2)

Number of tape channels

IP.NOISE (3)

Maximum decimal number of 12-bit bytes in a noise record on 7-track S and L tapes or 9-track conversion mode (S-format) magnetic tape. A record less than or equal to IP.NOISE is discarded.

IP.NOIS9 (5)

Maximum decimal number of 8-bit bytes in a noise record for packed mode on 9-track tapes. A record less than or equal to IP.NOIS9 is discarded.

IP.PTCN (13B)

Primary tape channel number. Used for internal purposes.

IP.RCYC (3R000)

Retention cycle (0-999) for calculating tape label expiration date when no retention cycle is given; 999 indicates permanent retention. The address field of the symbol definition should contain 3Rxxx where xxx defines retention cycle; leading zeros need not be written.

IP.RPE1 (12D)

Total decimal number of read parity retries on a single record (must be less than 60).

IP.RPE2 (8)

Decimal number of read parity retries accomplished by backspacing over the previous three records, then reading forward in an attempt to recover (IP.RPE2 must be less than IP.RPE1).

IP.TDEN (2)

Density for both label and data on 1/2-inch 7-track magnetic tape if not declared on LABEL or REQUEST statement: 0=556 bpi, 1=200 bpi, 2=800 bpi.

Tape scheduling options are as follows.

<u>Name</u>	<u>Bit</u>	<u>Value</u>	<u>Definition</u>
S.AUTO†	0	1	Enable automatic tape assignment according to LABEL or VSN specification.
S.URES†	1	1	Enable job scheduling based on job statement reservation.
S.PRES†	2	1	Enable prestaging features (the VSN preview of the P display).
S.2LBP†	3	0	Only ANSI labels are accepted and written.
		1	Two label formats (ANSI and 3000) are defined.
	4	1	Unused. <i>allow auto assignment to not ready error</i>
	5	1	Unused. <i>automatic unloading of tapes, if necessary, to make automatic assignment</i>
S.SCUL	6	1	Write-enabled, unlabeled tapes will be considered as usable for automatic assignment as scratch tapes.
S.SCEL†	7	1	Write enabled expired labeled tapes will automatically be considered for assignment as scratch tapes.
S.SCBL†	8	1	Write-enabled blank labeled tapes will automatically be considered as scratch tapes.
S.PREA	9	0	Give warning if tape job has no VSN information.
		1	Preabort such jobs.
S.OCJ†	10	0	Job initiation is based on tape drive availability; total demand cannot exceed number of drives logically available.
		1	Job initiation allows tape drive overcommitment.
S.UEOJ	11	1	Unless specified otherwise on REQUEST or LABEL statements, all tapes are unloaded at end of job.
S.PSON	12	1	Prestaging feature set on at deadstart time. This is equivalent to the STAGE ON typein.
S.TSEC	13	0	Tape security off. Specification of RING or NORING causes requested action.
		1	Tape security on, installation default active. Specification of RING or NORING causes requested action.
S.TRDO	14	0	Establishes installation default of NORING.
		1	Establishes installation default of RING.
	15-16		Unused.
S.NOOR	17		Operator cannot override VSN card.
S.DBUG	18		Enable label debug code (4LB,4LC).
	19-20		Unused.

†These bits are enabled in the default value of 2617B.

Tape scheduling options that can be selected by the installation are implemented by the use of conditionally assembled code. The bits in IP.TSG are tested at assembly time to determine the exact nature of the programs that comprise tape scheduling. For example, bit S.SCBL in IP.TSG governs the automatic scratch status of blank labeled tapes. If the bit is on, blank labeled tapes are considered scratch without operator intervention; if the bit is off, scratch status is not granted automatically.

The bits in IP.TSG can be divided into the three general categories of automatic assignment bits: prestaging bits, overcommitment bits, and miscellaneous bits.

#### AUTOMATIC ASSIGNMENT BIT

The installation can select automatic assignment by setting bit S.AUTO. With S.AUTO set on, a specific tape will be assigned automatically when the specific tape is mounted.

#### AUTOMATIC SCRATCH STATUS

Three other bits are related to automatic assignment. They are bits S.SCUL, S.SCEL, and S.SCBL. When set, each bit determines a specific type of tape to be considered automatically as a scratch tape. If all three bits are off, the only tapes treated as scratch are those specifically designated by the operator with the command SCRuu (where uu is the EST ordinal).

A job specifies \*MT or VSN = SCRATCH in the request for a scratch tape. If any automatic assignment is turned on (bit S.AUTO is set), the system will try to assign a scratch tape automatically to the job. The tape must be mounted on a ready unit with a write ring in place, it must also be designated as scratch as described above, and it must meet the following qualifications.

Tapes designated as scratch by the operator

Unlabeled tapes if bit S.SCUL is on

Tapes with expired labels if bit S.SCEL is on

Tapes with blank labels if bit S.SCBL is on

#### PRESTAGING BITS

##### Unit Reservation

Bit S.URES controls the necessity of job statement tape parameters, without which overcommitment and deadlock prevention are meaningless and prestaging will not function.

##### Prestaging

The prestaging option is assembled if bit S.PRES is set. If this option is on, a prestaging buffer is assembled in CMR; its length is N.VRNBUF\*6 (release value gives a 171B word buffer). Installations can change symbol N.VRNBUF in CMR to change the size of the buffer.

Complete VSN information cannot be obtained for jobs making internal tape requests or using tape file names repeatedly.

If bit S.PSON is on, it sets up CMR as if STAGEON had been typed after a normal deadstart. Deadstart recovery preserves the current setting of the STAGEON/STAGEOFF switch.

When bit S.PREA is set and prestaging is on (operator entered STAGE,ON or bit S.PSON set from deadstart tape), all jobs that specify tapes on the job statement but do not supply VSN information for all tape files requested are aborted.

## OVERCOMMITMENT BIT

Bit S.OCJI determines whether or not tape drives will be overcommitted. If the bit is off, the total number of tape drives required by all jobs executing at a given time (as determined by job statement tape parameters) cannot exceed the total number of tape drives at the installation. If bit S.OCJI is on, tape drives are overcommitted; the total tape requirements of executing jobs can exceed the total number of tape drives at the installation. Deadlock is prevented by an algorithm calculated each time a tape is assigned.

## MISCELLANEOUS BITS

### Two Label Processors

If, in addition to the ANSI label processor 4LB, 3000 Series (Y) labels are to be processed, bit S.2LBP should be on to allow use of the alternate label processor 4LC.

### EOJ Tape Unload

Bit S.UEOJ causes 1EJ to unload nonscratch tapes at end of job. If any problems are encountered when trying to unload the tape, such as tape not ready, the unload attempt is ignored. This differs from the SAVE (SV on REQUEST statement or X=SV on LABEL statement) unload processed by 1EJ; 1EJ issues a message that problems exist and continues trying until the operator types in GOuu.

### Operator Cannot Override VSN

With bit S.NOOR off, the operator can assign a tape with a VSN different from the VSN specified by the job; with S.NOOR on, a different VSN is not allowed.

### Label Debug

Bit S.DBUG controls debug code in 4LB and 4LC; use of this bit is not the normal mode of operation. This debug code produces many messages which show the calls to and returns from the label processors. Such messages may cause other more informative messages to be overwritten.

## OPTION DEPENDENCIES (IP.TSG)

The figures below show dependent bits. Each bit name shown cannot be turned on (turning it on will have no effect) unless all bit names below it are on. The three groups of bits are independent of each other. Miscellaneous bits (S.2LBP, S.UEOJ, S.NOOR, and S.DBUG) are independent of each other and of the bits shown below.

### Auto Assign Dependencies

S.SCUL	S.SCEL	S.SCBL
S.AUTO		

### Prestage and Overcommitment Dependencies

S.PSON	S.PREA	
S.PRES		S.OCJI
S.URES		

For example, S.PREA is dependent on S.PRES and S.URES but not on S.OCJI.

### Tape Security Dependencies

S.TRDO
S.TSEC

IP.WEC (0)

Hardware write error correction (applies only to 6250 cpi density). If IP.WEC is zero, the system allows certain types of single-track errors to be written that can be corrected when the tape is read (on-the-fly correction). This is the recommended setting because it provides efficient throughput, error recovery, and tape usage when writing GE data on media suitable for use at 3200 cpi or 6250 cpi.

When IP.WEC is one, the system invokes standard error recovery processing when an on-the-fly error occurs when writing a GE tape. The system erases the defective portion of tape, thereby reducing the amount of data that can be stored on the tape. Only tape which is suitable for recording at 6250 cpi should be used when this mode of operation is in effect.

**NOTE**

Users can override the installation parameter through the REQUEST and LABEL statement parameters EEC (enable error correction) and IEC (inhibit error correction). Refer to the NOS/BE 1 Reference Manual.

Refer to part III for a cross-reference listing showing the routines that reference each IPARAMS and CIOCOM symbol.

MISCELLANEOUS INSTALLATION VARIABLES

PASSWORD DEFINITIONS FOR THE SYSTEM DAYFILE

An installation may protect the system dayfile from unauthorized attaches by defining two passwords. The two passwords provide RD and XR permissions for the file. These passwords are defined as micros locally in both TDS (PL1A) and 1DF (PL1B).

The two micros are the following.

(XR permission) SDFXR

(RD permission) SDFRD

The default permissions defined in TDS and 1DF afford no protection for unauthorized attaches. An installation wishing to protect the dayfile must redefine micros SDFXR and SDFRD. The installation must consult a listing of TDS or 1DF to review the default definitions and to determine where new definitions must be inserted.

Installations wishing to change the SYSTEM dayfile passwords must make the same changes in both TDS and 1DF. When these changes are made they must be coordinated with an initial deadstart. When TDS catalogs the dayfile after an initial deadstart the passwords thus defined remain in effect.

MACROS TO DEFINE SPACING CODE ARRAYS FOR JANUS

580 printers equipped with programmable format control (PFC) make use of software defined arrays instead of format tapes to specify spacing codes. Two pairs of arrays (each containing a 6-lines-per-inch array and an 8-lines-per-inch array) are specified in PP routine 1IU. The first pair defined is the default pair, and is used for SC=0 and SC=1 on the ROUTE control statement. The second pair is the alternate pair, and is used for SC=2. Space exists for 61 installation-defined array pairs, corresponding to SC values 3 through 77. New arrays may be added by the use of the DPFC macro.

DPFC \*vcode\*,\*param\*

vcode Specifies the number of lines per inch for the array being specified

V6 6-lines-per-inch array

V8 8-lines-per-inch array

param Defines the actual array. This can be specified using letters A through L, O, and X. Letters A through L define channels 1 through 12, respectively. A indicates the beginning of the array, O indicates the end of the array, and X indicates no channel in that position. A letter must be specified for each line in the form.

Each array should also conform to the following criteria.

1. The array must begin with an A.
2. It must end with an O. This is an end-of-array terminator which is not counted as a line.
3. It must not be longer than 132 characters plus end-of-array terminator (6-lines-per-inch array) or 176 characters plus end-of-array terminator (8-lines-per-inch array).
4. It must contain each channel specified at least once in the array (L specifies the bottom of form).
5. Arrays must be specified in pairs; one 6-lines-per-inch array and one 8-lines-per-inch array. Either array may be specified first.

Example

```
*B DPFC.1
DPFC      *V6*,*AXBXCXDDEXFXGXHXIXJXXKBXCXLO*
DPFC      *V8*,*ABCDEFGHJKLO*
```

#### MACRO TO DEFINE THE EC PARAMETER ON THE JOB STATEMENT

The installation can change the syntax of the EC parameter field on the job statement through the OPTION macro in common deck 2VJCOM.

OPTION type,spec,mode,def1,base

type Specifies the parameter field name.

EC (required)

spec Specifies the format of the EC field.

0 EC can be specified without a value. A default value, def1, is supplied.

other A value is required with EC; EC by itself is ignored.

mode Specifies initial assignment of ECS field length at beginning of the job.

REDUCE ECS field length is not assigned.

other or  
omitted ECS field length is assigned.

def1 Specifies the default value (in multiples of 1000 octal) that is supplied when EC appears without a value. It should not exceed IP.MECS. It has no effect when spec is nonzero.

base Specifies the base of the value following EC.

DECIMAL Value is decimal.

other or  
omitted Value is octal.

On the released system, the OPTION call is as follows.

```
OPTION EC,1,REDUCE
```

This indicates that an octal value must follow the EC parameter on the job statement, and that the ECS field length will not be assigned at the start of the job.

#### CMR CONFIGURATION PARAMETERS (CMRIP)

The default values of the CMR configuration parameters are defined with the CEQU or CMICRO macros, so that an installation can insert all modifications at one given place. The CEQU and CMICRO macros are used to define variables conditionally. Since they are effective only if the variables have not been previously defined, any modifications should precede the release definitions.

Symbols can be defined by EQU or CEQU except for IP.SYSL1, IP.SLIB, IP.VER and IP.SYSE, which are micros and must be defined by CMICRO.

All the CMR configuration parameters are grouped together near the beginning of CMR.

General parameters should be tailored to suit the needs of each installation; default values are shown in parentheses.

Changes should be made as insertions after CMRIP.1 in an update of PL1A.

L.ELST	(20B)	Length of error logging status table. If defined as zero, no logging will be done.
L.EST	(40B)	Length of equipment status table ( $\leq 1000g$ ). Only RMS devices may be placed in the EST above 77g.
L.CST	(50B)	Length of channel status table. This parameter should be considered invariant.
L.INS	(0)	Length of installation table. Size, definition, and usage of an installation table is completely controlled by the individual site. No NOS/BE product set program makes reference to the installation table.
L.FNT	(2200B)	Length of file name table.
L.SEQ	(10B)	Length of the sequencer table. To use the sequencer, the value must be (2* number of jobs to be run)-2. If defined as zero, the sequencer cannot be used.
L.IDT	(40B)	Length of ID table. Must be nonzero multiple of 8.
LE.DFB00	(400B)	Size of system dayfile buffer may be less than 100g; if not, then it must be an even multiple of 100g.
LE.DFBXX	(77B)	Size of control point dayfile buffers.
LE.CERFB	(46B)	Size of hardware error file buffer. Subject to same limitation as LE.DFB00.
IP.ECSTP	(0)	Type of ECS. If zero, code for ECS I is executed; if nonzero, code for ECS II is executed.
N.BRKPT	(10B)	Maximum number of CPMTR breakpoints (for DEBUG only).
N.CP	(15)	Number of control points (1 to 15 decimal).
N.DEVICE	(3)	Number of controllers for allocatable devices; one for each 841, 844, or 885 disk pack controller (which may drive more than one disk pack unit). This parameter only has an effect if no RMS devices are assembled in the EST.
N.RBR	(3)	Number of record block reservation tables; normally one for each 841 unit, one for each 844-21 disk pack unit, two for each 844-41 double-density disk pack unit, and two for each 885 disk unit. This parameter only has an effect if no RBR cards are assembled.
N.RQS	(40)	Number of request stack entries.
N.VRNBUF	(20)	Number of entries in tape VSN buffer. Each entry is six words long and represents one line of job tape VSN information in the P display.
N.SPRPP	(1)	Number of PPs that are to be reserved for stack processor. In all cases, N.SPRPP must be at least one. If any dual access devices are defined, N.SPRPP must be at least two. The minimum value (1 or 2) should always be used if there are only seven or ten PPU's.
IP.FTHRL	(40B)	Default lower limit of free FNT entries. When the number of free FNT entries falls below this value, the system enters an FNT space critical condition. A multiple of 10g should be specified; the units position is truncated and assumed to be 0.
IP.FTHRU	(140B)	Default upper limit of free FNT entries. When the number of free FNT entries rises above this value, the system clears the FNT space critical condition. A multiple of 10g should be specified; the units position is truncated and assumed to be 0.
IP.DCT	(0)	Default controller type for 844 controllers.
	0	Half-track (7054) controller
	1	Full-track (7154) controller

IP.ELST	(0)	Initial value of error logging status byte. 0 is used for nondedicated mode, 3 is used for dedicated mode.
IP.SYSL1	(NOS/BE-1.4)	System label (up to 20 characters); the first character must be blank.
IP.VER	(RELEASE508)	System version identifier (up to 10 characters)
IP.SYSE	(12/15/79)	System generation date (up to 10 characters). These 10 characters are changed to type of deadstart and time when a deadstart recovery is done.
IP.SLIB	(CMRLIB)	Name of the library containing the segments and CMRTEXT corresponding to this CMR
L.APF	(64)	Length of APF table (2-word entries) $12 \leq L.APF \leq 8190$
N.SETS	(2)	Maximum number of device sets which may be mounted at any one time. The mounted set table (MST) will have a length of $N.SETS * L.E.MST$ . The range for N.SETS is 1 through 63.
N.VDDT	(3)	Maximum number of permanent packs for which jobs can be swapped out. The dismountable device table (DDT) will have a length of $N.VDDT +$ the number of RMS EST entries.

If more than SCERCNT errors are logged in the CERFILE within SCERINT seconds, the message SCR ERROR LIMIT EXCEEDED is displayed, and the system is placed in STEP mode.

If more than SCSECNT single bit data errors in central memory are corrected within SCSEINT seconds, an entry will be logged in the CERFILE.

If more than SCECCNT ECS errors occur within SSECINT seconds, an entry will be logged in the CERFILE.

Note that the default values for SCERINT and SCSEINT correspond to one hour intervals. Decreasing the value of SCSEINT causes the number of entries in the CERFILE to increase since an entry is logged each time a single bit error has occurred and the interval elapses.

#### INTERCOM parameters

L.ITABL	(19)	Length of INTERCOM multiplexer table
---------	------	--------------------------------------

#### System control point parameter

N.SBSYS	(0)	Maximum number of subsystems that may be defined in the subsystem control table. If zero, SCP code will not be assembled in CMR.
---------	-----	----------------------------------------------------------------------------------------------------------------------------------

#### Scheduler parameters

(Refer to Scheduling Parameters.)

L.ECSSWP	(3)	ECS swap mask. The value of this symbol forms a mask of bits that control job swapping when ECS is available.
----------	-----	---------------------------------------------------------------------------------------------------------------

<u>Bit</u>	<u>Meaning when set</u>
0	Swap any INTERCOM or graphics job to ECS at end of command.
1	Swap any batch job to ECS at end of time quantum or if waiting for memory.
2	Swap any INTERCOM or graphics job to ECS (overrides bit 0).
3	Swap any batch job to ECS (overrides bit 1).
4	Swap any job except MUJ to 819 (overrides bits 0-3).
5	Reserved.
6	Swap a job's direct access ECS and CM field lengths simultaneously. (This is effective only if IP.ECSW is nonzero; refer to ECS Installation Parameters.) A job's ECS is always swapped to mass storage.
7-11	Reserved.



If bits 0 through 4 are zero, only MUJ jobs and INTERCOM or graphics jobs at end of time quantum or waiting for memory will go to ECS.

Bit 4 should be set only if an 819 is available.

Bit 6 should be set only if the system has two separate channels to the mass storage devices that can hold swap files.

L.SCHJCA	(20B)	Length of job control area. Must be a multiple of 8. (Needs to be changed only if new classes are added.) L.SCHJCA can be redefined by means of the COMPASS pseudo-op SET.
L.SCHJDT	(400B)	Length of job descriptor table. Must be a multiple of 8.
AFL.BAS	(0)	Anticipated field length / 1000B used when INTERCOM is not up.
AFL.INT	(30B)	Increment to AFL / 1000B which is added when INTERCOM is brought up.

#### LOGICAL ID TABLE (LID)

The Logical ID Table (IDT) in CMR contains the mainframe Host ID (HID), the associated Logical IDs (LIDs) and the Physical Link ID (PID) of each currently linked mainframe of a multimainframe network. The last character of the HID in any multimainframe network must be a unique letter. One and only one HID can exist for a given mainframe IDT. Up to 58 logical IDs can exist. L.IDT must be nonzero and a multiple of 8. The default HID is MFA. The HID can be changed by the following.

\*INSERT,CMRIP.1  
HOSTID CMICRO , (xxx)                    where xxx is the desired HID.

Logical IDs can be added by the following.

\*INSERT,LID.1  
LID xyz                                    where xyz is the desired logical ID.

#### SYSTEM SECONDS

At end of job or when SUMMARY is executed, total system seconds is calculated and reported along with other job accounting in dayfile messages.

System seconds is expressed mathematically.

$$SS = CP*AW + IO*BW + CM*CW + EC*DW$$

CP is CPU A plus CPU B time in seconds.

IO is I/O time in seconds.

CM is central memory core seconds.

EC is ECS core seconds.

AW, BW, CW, and DW are installation selected weighting constants.

Central memory and ECS core seconds are similar and can be expressed mathematically.

$$(CP*EW + IO*FW) * FL$$

CP is CPU A plus CPU B time in seconds.

IO is I/O time in seconds.

EW and FW are installation selected weighting constants.



Terms and their sum (SS) are calculated by CP monitor when a PPU requests the M.ICE function, EX.SS subfunction. The elements CM and EC are calculated by the CP monitor routine PACKAGE each time central memory or ECS field length changes, or when system seconds is requested by a PPU.

The general format of the dayfile messages which show job accounting is the following.

\$CPA	Raw CPU A time in seconds	Weighted CPU A time
\$CPB	Raw CPU B time in seconds	Weighted CPU B time
\$IO	Raw I/O time in seconds	Weighted I/O time
\$CM	CM core seconds in kiloword seconds	Weighted CM core seconds
\$EC	ECS core seconds in kiloword seconds	Weighted ECS core seconds
\$SS		Sum of weighted terms
\$PP	Raw PP time in seconds	Today's date

PP time is reported although it is not added into the system seconds total.

Note that if IP.TCPUB is not equal to 4, then the figure referred to as raw CPU A time in seconds is itself an adjusted time. It is (seconds \*IP.TCPUB/4).

### WEIGHTING CONSTANTS

Weighting constants, mentioned under the System Seconds paragraph, have the following values in the system release version.

```

AW = 1
BW = 1
CW = .061035156250 (1000./40000B)
DW = .030517578125 (1000./100000B)
EW = .001
FW = .001

```

These values are not intended to be fixed or necessarily optimal for any individual installation. The weighting constants are released with these values so that installations can gather meaningful statistics in order to adjust the values at a later time. Note that CP and IO weighting constants AW and BW have the value of 1 in order to report actual time. CM and ECS core seconds weighting constants CW and DW cause the weighted values to be in 40K and 100K octal units, respectively.

Core seconds weighting constants EW and FW have the value of .001 and cause CM and ECS core second values to be in 1000 decimal or kilowords seconds. The ratio of 1 to 1 is used even though each installation must determine the best ratio for their job mix.

AW, BW, CW, and DW are defined at the end of CP.SS in CP MTR (near CRESCH.213), EW and FW are defined at the end of PACKAGE in CP MTR (near CRESCH.373).

Installations should not charge the user for PP time. A significant portion of PP time is system overhead not specifically requested or desired by the user. PP time will not necessarily be constant for the same job across several runs because PP time used by a job is dependent upon system activity and configuration. For example, PP time accrued by a user job will vary depending on the residency of the PP routines. Additionally, PP time charged will vary depending on the system activity when the job is run.

### TABLE STRUCTURES

Establishing a CMR for an installation requires inserting information about the CMR configuration parameters and tailoring the EST, RBR, and FLAW tables. Up to eight different CMR configurations, each with unique EST, RBR and FLAW tables, may be placed on the deadstart tape.

### EQUIPMENT STATUS TABLE

The EST may be tailored to any configuration by using the macros described in this section. Its size may be greater than or equal to the number of hardware units present in the configuration. However, it may not exceed 777 (octal) since an EST ordinal must be no more than nine bits. Since the first word of the EST cannot be used, the first equipment ordinal is 001. Only RMS devices may have an EST ordinal greater than 77B.

The CMR tables are defined by the TABLE macro. The sequence of the macro calls defines the sequence of the tables generated in CMR. This sequence can be altered by an installation, but the following constraints must be observed.

1. Origins of CST, EST, FNT, ITABL, DAT, RMSBUF, and STG tables must be located under 10000 (octal).
2. The RQS table must be located under 20000 (octal).

## EQUIPMENT CONFIGURATION

The EST macro defines the equipment in the configuration and the attributes associated with them. The actual parameters for the EST macro call are position independent and are given as a list of keyword=parameter fields separated by commas.

The EST macro causes a one-word EST entry to be constructed each time the macro statement occurs. For RMS devices, if no EST macro with the same type, channel 1, channel 2, and controller has been assembled, a DST and corresponding DAT entry will be constructed. If channel 2 is not blank, a second set of DST and DAT entries will be constructed for a dual access configuration that allowed for 841, 844, or 885. A DDT entry is also constructed for RMS devices, and an MST entry as well if MASTER is specified.

EST macro entries should be made at EST.1 in a PL1A update. The macro format is the following.

dt EST keyword=xxx, keyword=yyy. . .

dt Device type mnemonic; if dt is omitted, all parameters but ESTO are ignored.

Keywords include the following.

NAME Device identifier; any combination of up to 6 letters or digits (must be unique for each RMS unit; used to map RBRs).

ESTO EST ordinal (default is previous device count plus 1). The value must be larger than the previous device count.

CH Channel numbers; multiple channels are given CH=(C1,C2, . . .).

EQP Equipment number or display synchronizer number; multiple equipment numbers may be specified, such as EQP=(EQ1,EQ2,...).

UNIT Unit number; if an expander is being used with 844-21 disk packs, the unit number can be two digits. If 885 disks are used, unit numbers are two digits.

UNITS Number of units defined; default=1 (automatically incremented as more are defined) not allowed for disk.

MUX Multiplexer subtable index or ECS buffer number (if ECS link is used for multimainframe communication).

SN Setname; defines set membership of RMS device.

VSN Volume serial number of particular RMS device.

NF Used on master device only to specify the maximum number of permanent files that the PFD can hold (PFD will be slightly larger). Default is 32010 times the number of devices in the device set.

NM Used on master device only to specify the maximum number of members permitted in this set; default is equal to number of members in EST plus 5.

FC Applies to printers, specifies an optional forms code (equivalent to operator's DSP type-in); value equals any two alphanumeric characters.

EC Applies to printers, specifies external character set (type of print train on the printer); values include the following.

A4	ASCII	48-character set
B4	BCD	48-character set
A6	ASCII	64-character set
B6	BCD	64-character set
A9	ASCII	96-character set

MOD Specifies device attributes; multiple attributes are given as MOD=(value1, value2, . . .) where value is one of the following keywords.

For all devices  
OFF

For RMS devices, values are

FREE	
SYS	System files may reside on device.
PF	Permanent files may reside on device.
QUE	Queue files may reside on device.
SHAR	Drive is shared between mainframes (applies only to 844s).
IDLE	Drive is initially idled.

For tapes, values are

ATS Specifies 67x tape subsystem.  
GCR Used in conjunction with ATS parameter to indicate an ATS unit with 6250-cpi density capability.  
BID Used in conjunction with the MTS parameter to indicate Block ID capability for 66x units (for example, MOD=(MTS, BID)).  
MTS Specifies 66x tape subsystem.  
MMTC Specifies 65x tape controller.  
6684 Specifies channel converter.

For printers

PFC Printer is equipped with programmable format control (applies only to 580 printers).

MASTER Specifies attributes of set for which this is the master device; multiple attributes are given as MASTER=(M1,M2, . . .)

SYS System files may reside on set.  
PF Permanent files may reside on set.  
QUE Queue files may reside on set.  
SCR Scratch files may reside on set (system default set).

TYPE Specifies the controller type for the previously specified channel and equipment numbers (in one-to-one correspondence if multiple parameters are included); multiple types can be specified as TYPE=(TY1,TY2,...).

7054 Specifies 7054 controller.  
7154 Specifies 7154 controller.  
7155 Specifies 7155 controller.

This entry is used only for 844 and 885 devices; it is ignored for other devices. If TYPE is not defined, 7155 is the default value for 885 disks, and the value of IP.DCT determines the default for 844 disks. If TYPE is not defined for the second, third, or fourth access of a multiple access disk, the last defined type is used.

**NOTE**

An attempt to redefine TYPE in a subsequent EST with identical CH and EQP attributes is ignored and the original definition is used.

SO Specifies the stack request scheduling option for RMS devices.

SEL Specifies optimization of requests within each unit and unit selection scheduling.  
SEEK Specifies overlap seek optimization in addition to SEL optimization.  
FIFO Sends stack requests to the device driver in the order received.

The default value is SEEK. This parameter is not allowed for 819 devices.

The device type may be any of the following.

AH	819 disk drive	LQ	Line printer (512)
AJ	885 disk drive	LR	Line printer (580-12)
AM	841 disk drive	LS	Line printer (580-16)
AY	844-21 disk drive	LT	Line printer (580-20)
AZ	844-41 disk drive	MT	Magnetic tape (657, 667, 677)
CC	6683 satellite coupler	NT	Magnetic tape (659, 669, 679)
CR	Card reader (405)	CP	Card punch (415)
CS	791 LCC mux	DS	Console display
CX	ECS link	SC	6673/6674 wide-band mux
ED	Distributive data path (DDP)	DC	6671 low-speed mux
FE	255x Front End	YC	6676 low-speed mux

**NOTE**

The shared attribute is not specified via the EST macro for RMS controllers. This is not necessary, because the software assumes that any controller may be shared.

The AH device type can have only the attributes ON, FREE, PF, IDLE, and OFF. It cannot have the system device attribute (SYS) nor can it be a MASTER device. Also, it must be a member of a public set.

Parameters defining hardware configuration are required (CH, EQP, UNIT). RMS devices also require NAME, VSN, and SN, or only NAME if MOD=FREE. Do not use BSSZ to create spaces in the EST; use the ESTO parameter.

**CAUTION**

A mixed 844 RMS controller configuration (7154 full-track controllers and 7054 half-track controllers) sharing 844 drives is allowed but must be used with caution. This configuration must be operationally limited to data recorded in half-track mode to prevent serious performance degradation. This degradation occurs if full-track recording operations are performed through the 7054 half-track access.

All numeric parameters are assumed octal unless otherwise specified. Multiple parameters must be enclosed in parentheses. Channel numbers must be set up in order of precedence, except for RMS devices where channels should be specified in ascending numerical order.

**NOTE**

Before a continuation card can be used, information must be punched through column 72 of the previous card and the continuation card must begin with a comma in column 1.

In the following examples, the device type begins in column 1, the macro name in column 11, and the parameter string in column 18.

```
AY      EST      NAME=844D,CH=2,EQP=0,UNIT=3,ESTO=10,MOD=(OFF,QUE,SHAR),
,VSN=SHARIO,SN=IOQUES,MASTER=(QUE,SCR),NF=500,NM=3
```

The above creates a master device for set IOQUES with VSN SHARIO. This pack is shared and holds queue files; the controller type has either been previously specified for channel 2 and equipment 0, or the system default defined in CMR will be used.

```
AZ      EST      NAME=844DBL,CH=32,EQP=0,UNIT=7,ESTO=127,MOD=FREE,VSN=DB
,LDEN,SN=HICAP,MASTER=SCR,NM=2,TYPE=7154
```

The above creates a master device for set HICAP with VSN=DBLDEN. This double-density pack can be used as a scratch pack. It is at EST ordinal 127 and is connected as unit 7. The controller type is set to full track if no prior definition has been made for channel 32 and equipment 0.

```
AY      EST      NAME=844A,CH=7,EQP=5,UNIT=0,MOD=FREE
MT      EST      CH=(11,4,5),EQP=7,UNIT=0,UNITS=8D,MOD=(MTS,BID)
```

The above creates units 0 through 7 (7-track 667s) with 6684 data channel converter.

```
DS      EST      CH=10,EQP=7
NT      EST      CH=(13,12),EQP=5,UNIT=2,UNITS=4,MOD=(MTS,BID),ESTO=42
```

The above creates units 2 through 5 (9-track 669s) at EST ordinals 42 through 45.

```
CX      EST      MOD=OFF,MUX=1
```

The above creates an ECS Link using ECS buffer 1 (MUX=1) for multimainframe communication.

The OFF designation for RMS devices causes all record block assignment to be prevented. During deadstart, OFF drives are still checked and labeled. Only IDLE drives are ignored.

Devices in a device set having the system set attribute cannot be designated as shared (that is, the system set cannot be shared).

Parameters SN, VSN, NM, and NF are only considered at the time the device set is initialized; changing them later without reinitializing the device set has no effect. The value of these parameters should be chosen with a view toward future expansion.

Changing set attributes [specified via the MASTER=(M1,M2 . . .) parameter] does not require set initialization. Set attributes can be changed on any load (I.L) deadstart.

Device attributes PF and QUE can only be changed if the device set is initialized [MOD=(PF,QUE)].

Device set attribute SYS can be changed on any 1.L type deadstart.

EST and/or RBR ordinals may be changed without a requirement to reinitialize the set. (All disk resident system tables are CMR independent.)

**NOTE**

To avoid degradation in system throughput, the system resident device should be on a channel separate from any other equipment. In addition, SYSTEM and/or PFD residency should be placed on double-ranked channels (24B-33B) in CDC 6000 or CDC CYBER 70 series mainframes with more than 10 PPUs or 12 channels.

Because certain tables (such as PFD, PFC, and DAM) on a device set are modified under the protection of the stack request interlock (which is issued on a controller basis), any set can interfere with the performance of other sets on the same controller. If any of the sets sharing one controller are public sets, system degradation may result.

Equipment (controller) numbers for 841 devices must be 4, 5, 6, or 7.

Equipment (controller) number for an RMS controller must be zero; thus, a channel cannot have two RMS controllers. No testing has been done with any other equipment on the same channel with an 844 or an 885.

If 844 full-track recording mode is used on a 20-PPU system other than CYBER 170 series model 720, 720, 750, or 760, disk revolutions may be lost when a PPU conflict occurs. The conflict occurs when the stack processor partner PPU is executing I/O instructions. Although lost revolutions may occur during conflict, performance degradation is minimal. CYBER 170 model 7xx mainframes support full-track recording without lost revolutions.

An expander can be used with single density 844-21 drives only. A 6-bit numbering scheme is used for 844-21 and 844-41 units. A site with no expander uses 00 through 07 as in the past.

A site with an expander on each port uses drives 00 through 07 as the first rank (first drive on each expander), 10 through 17 as the next rank, and so on. Thus, drive 35 is the fourth drive connected to the sixth port of the controller (expander five).

Disk pack entries assembled into the EST for the installation CMR should be designated as idle, allowing the operator to mount public or private devices as required.

Dual access 841/844/885 allows simultaneous data transfer to two members of a group of 841, 844, or 885 mass storage units, where:

- For 841, the group must be connected to two 3553-1 controllers and each unit must have dual access option 10163 installed.
- For 844-21, the group must be connected to two controllers (either 7054 or 7154 types) that can be loaded with a compatible version of the controlware.
- For 844-4x, the group must be connected to two controllers (7054, 7154, or 7155 type) that can be loaded with a compatible version of the controlware.
- For 885, the group must be connected to two 7155 controllers.

There is no advantage in designating dual access if one or both of the controllers is being actively shared by two mainframes.

If the dual access configuration uses one single channel coupler, FCO CA32618 must be installed in both controllers.

Usefulness of the dual access 841/844/885 feature, relating to the improvement of efficiency, mainly depends on the type of job mix and the number of units in the group. Even though the number of units is three, under circumstances where relatively large data transfers are expected to each of the units simultaneously or randomly, a fair improvement of throughput can be expected. Conversely, if the job mix is compute-bound and the total of RMS processing time is less than the elapsed time, or where the delay of stack requests costs nothing because of multiprogramming, no improvement is expected regardless of the number of units.

If IPARAMS symbol IP.CSET is equated to IP.C63, the data channel converters used must all be 6681s, all 6684-Is or 6684-IIs used as 6681s.

If IPARAMS symbol IP.CSET is equated to either IP.C64.1 or IP.C64.2, the data channel converters used must be all 6681s, all 6684-IIs or all 6684-Is used as 6681s.

Equipment numbers of 657 and 659 magnetic tape controllers must be 4, 5, 6, or 7. If 657 or 659 tape units are to be used for deadstarting, they must be configured on channels 0, 12, 13, 32, or 33.

Equipment number for the 667 and 669 (MTS) controller must be 0. Channel 0 cannot be used for the MTS (66x) or ATS (67x) tape subsystems. ATS (67x) controller equipment numbers can range from 0 through 7.

Each 6000 channel can have only one 6681 or 6684 channel converter. This restriction does not exclude the use of a 6000 type controller on the same channel with the 6681 or 6684 converter.

If one of the channels is channel zero, it must appear as the first channel.

The channel must not include the high-order bit (40 octal) for the 6684; this is a function of the 6684 parameter.

6683 couplers cannot share a channel with other equipment. Use either a dedicated channel (CC) for the 6683, or turn all other equipment on the 6683 channel logically OFF.

### RECORD BLOCK RESERVATION TABLE (RBR)

Each mass storage device is represented by at least one entry in the RBR. Several RBRs can be generated for a single device, each describing a unique area on the device. Each entry includes a two-word header and a variable length bit table.

The first word of each RBR header contains a 6-bit allocation style code supplied as a parameter to the RBR macro when the CMR is assembled at an installation. Unique allocation style codes for each RBR can be set by the installation; this code can be used to direct a file to the RBR with a specific RB size or recording technique.

An RBR table is a single bit string of variable length, up to a maximum of 4095 bits. Each bit represents the availability of the corresponding record block (RB); the number of PRUs per RB is constant throughout the table. On the first RBR of a system device, the RB size is limited to 80 PRUs maximum.

The RBR macro is defined in an order-independent parameter format where the parameters consist of keyword=value. Keywords and values are in the following description. All numeric values are assumed decimal unless otherwise specified. RBRs are added to CMR by inserting RBR macro statements into RBR.1 in the following form.

name	RBR	keyword=xxx,keyword=yyy,...
	name	RBR name. This name must be specified in the EST macro.
	COUNT=rblocks	Number of record blocks in this RBR (required parameter)
	PRURB=prus	Number of physical record units per record block. If not specified, the default value is used.
	ALLOC=style	Allocation style. If not specified, the default value is 0.
	DEFAULT=number	Default file assignment. Default file assignment is inhibited if number is 0. Default files are assigned if number is any other value or if DEFAULT=number is not specified.
	MAXRB=max	Upper threshold for DAM processing on a shared device (ignored for an unshared device). SPM returns space to the DAM when the number of locally available RBs reaches max. The default value is 1000B.
	MINRB=min	Lower threshold for DAM processing on a shared device (ignored for an unshared device). SPM obtains space from the DAM when the number of locally available RBs decreases to min. The default value is 40B.

Example for 844-21

```
844A    RBR    COUNT=3232
```

Private devices are configured by LABELMS, independent of the RBR declarations in CMR.

RMS drives configured for private device usage must be defined to have no more than eight RBR table entries per device. The RB size specified must be not less than 1/32 of the PB (physical block) size and not greater than 32 times the PB size.

The 844 physical block (PB) size changed from 56 to 112 PRUs/PB with the introduction of the 844 double density feature at PSR level 430. For 844 pack upward compatibility from earlier systems to level 430 or later systems, the RB size for 844 devices is restricted to the following values. These RB sizes apply for upward compatibility regardless of whether the installation is using the 844 double density feature.

RB sizes less than or equal to 56 are 4, 7, 8, 14, 28, and 56

RB sizes greater than 56 may be

$$(2n-1)*56+1 \leq \text{RB size} \leq 2n*56 \text{ where } n=1,2,\dots,32$$

Example

```
For n=1, 57 < RB size < 112  
For n=2, 169 < RB size < 224
```

Consecutive sections of any device must be specified by consecutive RBR statements. The installation should set up the RBR table entries for drives to be used for private devices such that the maximum number of RBRs for any device is also the maximum number that will be used by any private device; and the total bit table size (controlled by the RB size and number of RB) for the drive should also be the maximum required by any private device.



Example for 844-41

844F RBR COUNT=3232  
 844F RBR COUNT=3232

<u>Device</u>	<u>Mnemonic</u>	<u>PB Size</u>	<u>Number of PRUs in Default RB</u>	<u>Corresponding Count</u>
844-21 pack	AY	114	57	3232
844-41 pack	AZ	114	57†	3232
819 unit	AH	160	160††	4030
885 unit	AJ	320	160†	3368

Every RMS device is logically divided into groups of PRUs called physical blocks (PBs). The number of PBs per device must not exceed 4095.

1 PRU = 1 sector = 64 CM words

An RB represents the minimum amount of disk space that can be assigned to a file. RB size need not be equal to PB size.

If the RB size is less than the PB size but does not divide it exactly, disk space is lost. This is because an RB assignment will not be made starting in the middle of a PB if that RB would then overlap to the next PB. Instead, the RB begins on the next PB boundary and the remaining PRUs in the current PB become unavailable.

Example (PB=114 PRUs)

<u>RB Size (PRUs)</u>	<u>RBs/PB</u>	<u>Unused PRUs/PB</u>
14	8	2
56	2	2
57	2	0
110	1	4

If the RB size is greater than the PB size but is not an exact multiple of it, disk space is lost. Since an RB assignment will not be made starting in the middle of a PB and since each RB is greater than the PB size, it follows that all RBs will start on a PB boundary and that unused PRUs in the last PB of an RB become unavailable.

Example (PB=114 PRUs)

<u>RB Size (PRUs)</u>	<u>PBs/RB</u>	<u>Unused PRUs in Last PB</u>
169	2	59
222	2	6
224	2	4
281	3	61
1120	10	20

FLAW TABLE

The FLAW macro and the deadstart FLAW inputs have been changed to use physical addresses, in the same format as LABELMS. Numbers are assumed octal unless otherwise specified. FLAW macro entries should be inserted at FLAW.I in the following format.

```

name    FLAW    (string)
        name: RBR name
        string: Physical address of flaw (must be in parentheses)
  
```

† Requires two RBRs to fully describe disk space.  
 †† For the 819 unit, PRU/RB cannot be changed from the default of 160. For the 819 unit, 1 PRU = 64 CM words = 1/8 sector.

The formats of the flaw strings are as follows.

<u>Device</u>	<u>Format</u>	<u>Values</u>	<u>Parameters</u>
841	Txx,Cyyy,Szz.	0 ≤ xx ≤ 23g 0 ≤ yyy ≤ 307g 0 ≤ zz ≤ 15g	T Track number C Cylinder number S Sector number
844-21	Txx,Cyyy,Szz.	0 ≤ xx ≤ 22g 0 ≤ yyy ≤ 632g 0 ≤ zz ≤ 27g	
844-41	Txx,Cyyy,Szz.	0 ≤ xx ≤ 22g 0 ≤ yyy ≤ 1466g 0 ≤ zz ≤ 27g	
819	Txx,Cyy,S0.	0 ≤ xx ≤ 10g 0 ≤ yy ≤ 633g	
885	Txx,Cyy,S0.	0 ≤ xx ≤ 47g 0 ≤ yy ≤ 1512g	

For all devices, the sector parameter (S) may be specified in the following form.

Sbbb-eee

bbb: Beginning sector number  
eee: Ending sector number

For an 819, the sector parameter(s) need only be specified as S0. NOS/BE 1 accesses one PB (track) at a time.

Example

844A            FLAW            (T12,C421,S24-26)

FLAW statements for the same device must be contiguous. Devices to be flawed must have SN and VSN specified in their EST macro call.

For the 844-21 (AY) and the 844-41 (AZ), NOS/BE 1 reads the utility flaw map to obtain the disk flaws. This is done when the device is a member of a public set being initialized or modified during deadstart, or when the device is labeled by LABELMS. Refer to section 17 for a description of Factory Format Support.

#### COMPUTING PHYSICAL ADDRESSES FOR CONVERTING RB NUMBERS

If the cylinder/track/sector of a flaw is not known, but the RB number and the PRU number are known, the following diagrams and formulas may be used to convert the RB and PRU to the physical address (cylinder/track/sector) of the flaw.

First, compute the PB (physical block) and SPRU (standard PRU) from the RB and PRU. If the device has one RBR and an RB size equal to the PB size, then PB = RB-1 and the SPRU is the same as the PRU. Otherwise, consult nondefault RB sizes as shown.

819            PB=RB

Cylinder    (PB-1)/10  
Headgroup   PB-1-cylinder\*10 (Headgroup is synonymous with track; resulting values should be truncated)  
Sector       PRU/8 (since there are eight PRUs per 819 sector)

841    Cylinder is PB/5

Let

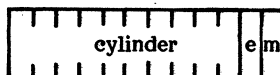
CPB (cylinder PB) be remainder of PB/5  
CS (cylinder sector) be CPB\*56+SPRU

Then

Track = (CS/7) mod 20  
Sector = (remainder of CS/7)\*2+CS/140

844-21, 844-41, and 885

PB number



Half track

m = cylinder position

e = even or odd PRU number within the PB

0 for even  
1 for odd

Cylinder = bits 2 through 11 of the PB number

PBS = PB size in PRUs  
RBS = RB size in PRUs  
TRS = track size in PRUs

Let

$CS \text{ (cylinder sector)} = PBS * m + SPRU + (\text{integral number of } SPRU / RBS)$

Then

$Track = CS * 2 / TRS$   
 $Sector = (\text{remainder of } CS * 2 / TRS) * 2 + e$

#### Full-track

Cylinder = bits 2 through 11 of the PB number

Let

EM = bits 0 and 1 of the PB number  
 $CS = PBS * EM + SPRU + SPRU / RBS$

Then

$Track = CS / TRS$   
 $Sector = \text{remainder of } CS / TRS$

#### Nondefault RB sizes (PB size $\neq$ RB size).

To compute the PB and SPRU for nondefault RB sizes, do the following.

1. When the RB size is larger than PB size, one RB fits in an integral number of PBs; let this integral number be the factor.

$PB = (RB - 1) * \text{factor} + PRU / (PB \text{ size})$   
 $SPRU = \text{remainder of } PRU / (PB \text{ size})$

2. When the RB size is smaller than the PB size, one PB contains an integral number of RBs; let this integral number be the factor.

$PB = (RB - 1) / \text{factor}$   
 $SPRU = PRU + (\text{remainder of } RB - 1 / \text{factor}) * RB \text{ size}$

3. In both cases, if the RB was not in the first RBR of the device, determine the starting PB of the RBR from the CMR assembly or dump, and add this value to the PB.

#### CMR EQUIPMENT CONFIGURATION EXAMPLES

Example modifications to CMR for installation of the following equipment, including 819 RMS devices (CDC CYBER 170 Model 176 only).

415 card punch on channel 5, equipment 4

405 card reader on channel 12, equipment 4

Console on channel 10, controller 7

Two 580 PFC printers on channel 11, equipments 6 and 7, print train BCD, 64-character set

Sixteen magnetic tape units on channels 5, 11, 12, and 13 with 6681 converter, equipment number 5, units 0 through 17B

Three 9-track magnetic tape units on channel 7 with 6681 converter, equipment 7, units 0, 1, and 2

Two 844 units (one available as a nonshared private device and one available as the system resident device and master device of the public set containing 819 devices)

Four 819 units configured on FLPP channels 4, 5, 6, and 7 (2 by 4 access), equipment 7, units 0 through 3 (available for scratch and permanent files)

```
*INSERT      EST.1
AY           EST      NAME=844A,CH=4,EQP=0,UNIT=0,MOD=(SYS,PF,QUE),SN=PUBLSET
,,VSN=SYS000,MASTER=(SYS,PF,QUE,SCR)
CP          EST      CH=05,EQP=4
CR          EST      CH=12,EQP=4
LR          EST      CH=11,EQP=6,MOD=PFC,EC=B6
LR          EST      CH=11,EQP=7,MOD=PFC,EC=B6
DS          EST      CH=10,EQP=7
MT          EST      CH=(13,12,11,5),EQP=5,UNIT=0,UNITS=16,ESTO=10
NT          EST      CH=7,EQP=7,UNIT=0,UNITS=2
AY          EST      NAME=844B,CH=4,EQP=0,UNIT=1,MOD=(FREE,IDLE)
AH          EST      NAME=819A,CH=(4,6),EQP=7,UNIT=0,MOD=PF,SN=PUBLSET,VSN=S
,YS002
AH          EST      NAME=819B,CH=(4,6),EQP=7,UNIT=1,MOD=PF,SN=PUBLSET,VSN=S
,YS003
AH          EST      NAME=819C,CH=(4,6),EQP=7,UNIT=2,MOD=PF,SN=PUBLSET,VSN=S
,YS004
AH          EST      NAME=819D,CH=(4,6),EQP=7,UNIT=3,MOD=PF,SN=PUBLSET,VSN=S
,YS005
*INSERT      RBR.1
844A        RBR      COUNT=3232
844B        RBR      COUNT=3232
819A        RBR      COUNT=4030,PRURB=160
819B        RBR      COUNT=4030,PRURB=160
819C        RBR      COUNT=4030,PRURB=160
819D        RBR      COUNT=4030,PRURB=160
```

Example CMR modifications for installation of the following equipment.

415 card punch on channel 5, equipment 4

405 card reader on channel 12, equipment 4

Console on channel 10, synchronizer 7

Two 580 PFC printers on channel 11, equipments 6 and 7, print train ASCII, 64-character set

Sixteen magnetic tape units on channels 5, 11, 12 and 13, 6681 converter, equipment number 5, units 0 through 17B

Three 9-track magnetic tape units on channel 7, 6681 converter, equipment 7, units 0, 1, 2

Five 844 units, one available for shared private device, one available for nonshared private device, and two members of system device set designated as queue devices; channel 4, equipment 0, and units 0-4. All are accessed with a full-track controller.

Two 885 units and two 844-41 units, all accessed by a 7155 controller on channels 0 and 3.

```

*INSERT      EST.1
AY           EST      NAME=844A,CH=4,EQP=0,UNIT=4,MOD=(SYS,PF),SN=PUBLSET,VSN
,=SYS000,TYPE=7154
CP           EST      CH=05,EQP=4
CR           EST      CH=12,EQP=4
LR           EST      CH=11,EQP=6,MOD=PFC,EC=A6
LR           EST      CH=11,EQP=7,MOD=PFC,EC=A6
DS           EST      CH=10,EQP=7
MT           EST      CH=(13,12,11,5),EQP=5,UNIT=0,UNITS=16,ESTO=10
NT           EST      CH=7,EQP=7,UNIT=0,UNITS=2
AY           EST      NAME=844X,CH=4,EQP=0,UNIT=0,ESTO=40,MOD=(FREE,IDLE,SHAR
,)
AY           EST      NAME=844B,CH=4,EQP=0,UNIT=1,MOD=FREE,IDLE)
AY           EST      NAME=844C,CH=4,EQP=0,UNIT=2,MOD=QUE,SN=PUBLSET,VSN=SYS0
,03
AY           EST      NAME=844D,CH=4,EQP=0,UNIT=3,MOD=QUE,SN=PUBLSET,VSN=SYS0
,04
AJ           EST      NAME=885A,CH=(0,3),EQP=0,UNIT=0,MOD=(FREE,IDLE),TYPE=7155
AJ           EST      NAME=885B,CH=(0,3),EQP=0,UNIT=1,MOD=(FREE,IDLE),TYPE=7155
AZ           EST      NAME=844E,CH=(0,3),EQP=0,UNIT=0,MOD=(FREE,IDLE),TYPE=7155
AZ           EST      NAME=844F,CH=(0,3),EQP=0,UNIT=1,MOD=(FREE,IDLE),TYPE=7155
*INSERT      RBR.1
844A         RBR      COUNT=3232
844X         RBR      COUNT=3232
844B         RBR      COUNT=3232
844C         RBR      COUNT=3232
844D         RBR      COUNT=3232
844E         RBR      COUNT=3232
844E         RBR      COUNT=3232
844F         RBR      COUNT=3232
844F         RBR      COUNT=3232
885A         RBR      COUNT=3368
885A         RBR      COUNT=3368
885B         RBR      COUNT=3368
885B         RBR      COUNT=3368

```

#### DEADSTART INSTALLATION PARAMETERS (DSLCOM)

The release value default parameters in the Operator Option Matrix are determined by a condition micro in the deadstart parameters common deck DSLCOM. The default CMICRO OPTDF determines the initial values for the options. Deadstart option defaults can be changed by inserting, at DSLCOM.12, either a MICRO or a CMICRO. Examples to set recover I/O queues to NO and to initialize ECS are as follows.

```

      OPTDF  CMICRO  ,NNNNIN
or  OPTDF  MICRO   1,,$NNNNIN$

```

The default of the options appear from left to right in the MICRO. To change a default value, insert a micro (named OPTDF) with the desired changes at DSLCOM.12.

The MICRO as released includes the options following (left to right in the micro string).

- 1.N Reload libraries
- 2.N Recover I/O queues
- 3.N Validate user sets
- 4.N Equipment changes

The following two values may or may not be defined. Their assembly is governed by the IPARAMS symbols IP.ECSB and IP.SRMS.

- 5.I Initialize ECS
- 6.N First mainframe to deadstart

All symbols described below are defined in the common deck DSLCOM. Default values are shown. A 65K memory is assumed. Central memory usage by deadstart may be modified by changing the symbol values at DSLCOM.12 on PL1A. Most symbols are keyed from a symbol defining an adjacent area, and all depend on the value of the symbol BASE. For example, if a 131K system is to have an unusually large CM resident library, it may be necessary to set the origin address of IRCP (IRADR) to a higher value redefining BASE to any arbitrary address in the middle of CM. For central memory larger than 65K, ample space is available to enlarge both the CM resident area and the RBT area.

BASE	CEQU	124000B	Location from which origins of other areas are keyed
IRADR	CEQU	BASE-26000B	Absolute origin address of IRCP (defines size of IRCP)
TBUF0	CEQU	BASE-42120B	Lowest data block used by IRCP
CMRSIZE	CEQU	16000B	Number of words in CMR to be saved for recovery purposes
DSPLCHAN	CEQU	10B	Display channel number
DSPLCTLR	CEQU	7	Display controller number
ROCKCNT	CEQU	10B	Retry count for tape parity error
DRIVBFL	CEQU	23000B	DRIVBUF length

The following dependencies and constraints exist and may be helpful in making changes.

1. The central memory resident libraries/programs must not extend past IRADR or loading cannot complete. If they do, BASE must be redefined.
2. Deadstart recovery attempts to recover the INTERCOM user tables which are located after the CM library and are typically about 5000g CM words. Deadstart will recover all user tables before TBUF0 and abort all users with tables above TBUF0.
3. IRCP must not be larger than BASE - IRADR or it will overlay DRIVBUF. An assembly error will occur if an attempt is made to generate an IRCP larger than the current value of BASE - IRADR.

If ECS is to be defined (IP.ECSB nonzero), the deadstart parameter IRADR must be changed. A value of BASE-32000g is suggested. This must be done in addition to adjusting BASE, depending upon machine central memory size.

4. DRIVBUF contains copies of IRP (the RMS driver), the RMS driver overlays (one for each device type), 885 BC controlware, 844 BC controlware, and 66x BC controlware. To save CM space in deadstart, any of these drivers or controlware that are not needed by an installation can be removed from the deadstart tape and the size of DRIVBUF shortened appropriately. (Note that the controlware packages are about 3200g CM words each, except 885 controlware which is about 5500g CM words, and the 819 subsystem which is less than 1000g CM words.) The length of this buffer is controlled by the value of DRIVBFL.
5. When the old CMR is saved for recovery, the number of words to be moved is determined by the DSLCOM symbol CMRSIZE. This value includes all the CMR tables recovered by deadstart. If IP.ECSB = 0, the JDT (job descriptor table) is the highest table recovered and CMRSIZE should correspond to the start of CP.MTR in CMR. If IP.ECSB ≠ 0, the empty page stack is the highest table recovered and CMRSIZE should correspond to the start of the ECS parameter table (symbol T.ECSPRM in CMR). If CMR is larger, CMRSIZE must be redefined.
6. For load-type deadstart (1.L), the RMS flaws are maintained in a backward list starting at machine size - RBTSIZE. (RBTSIZE is a DSLCOM symbol which defines the maximum size of the RBT area needed for the system file.) The RMS flaws are passed to post deadstart in a list at the end of control point zero. This list must not extend past IRADR. If it does, a warning message is displayed and the overlapping flaws are discarded.

7. It is strongly recommended that the BASE origin be as high as possible, up to 377777g. Execution of IRCP is not guaranteed when loaded above 377777 minus IRCP length. The following formula can be used as a guideline.

Let

MAXRBT = Maximum length of RBT area expected to be used by the installation. (Deadstart recovery expects the RBT area to be intact. If the RBT delimiter cannot be found in the last 20000B CM words, recovery is not possible.)

DOSIZE = DRIVBFL + size of OPCOM buffer (DOSIZE = 24000B on the unconfigured deadstart tape).

BUFSIZE = Maximum of (17620B, CMRSIZE). 17620B is the combined size of buffers STLBUF, MTRBUF, DSDBUF, and CMBUF. CMRSIZE is the DSLCOM symbol used for saving CMR.

Then

BASE = Machine size - MAXRBT - DOSIZE - BUFSIZE

#### Example

Suggested values depending on the size of central memory are as follows.

<u>Central Memory</u>	<u>CMRSIZE</u>	<u>MAXRBT</u>	<u>BASE</u>
65K	16000B	10000B	124000B
98K	22000B	20000B	223000B
131K	22000B	20000B	323000B
196K	22000B	20000B	523000B
262K	22000B	20000B	723000B

8. Increasing the value of BASE may restrict the capability to perform checkpoint recovery deadstarts with memory degraded.

#### PERMANENT FILES AND DEVICE SET INSTALLATION

Under NOS/BE, all RMS devices are grouped into device sets. Each of these device sets has a setname (SN). Within the device set, the device has a unique name and a volume serial number (VSN). The setname and volume serial number are recorded in the RMS label of the device.

Device sets can be used in one of two ways: as a public set or a private set. Private sets are device sets which are typically used by a subset of jobs and, therefore, their availability may be determined by whether they are requested by any jobs. Public sets are defined and maintained by the installation and are public throughout the running of the system. The usage of the public sets must be additionally qualified by the application of public set attributes to the device set. The possible attributes are the following.

- System set attribute determines that all permanent files of ID=SYSTEM are to be resident within this device set. In addition, this device set contains all the system devices.
- Scratch set attribute determines that scratch file assignment is to be to the applicable device sets.
- Permanent file default set attribute determines that default permanent file assignment is to be to the applicable device set.
- Queue set attribute determines that queue (input and output) file assignment is to be to the applicable device set.

The scratch set attribute is the only one which can be applied to more than one public set at the same time; otherwise, any combination of device set attributes can be applied to a public set. The maximum number of public sets is four and all attributes must be assigned. A device which is currently a member of a public set is called a public device. A device which is currently a member of a private set is a private device.

Within a device set, certain device attributes can be applied to the devices within the device set. These attributes qualify how a particular device is to be used while a member of the device set. Specifically, the device attributes include the following.

- Master device attribute defines the specified device on which the disk resident tables are to be resident.
- Permanent file device attribute defines that permanent files can be assigned to the specified device.
- Queue file device attribute defines that queue files can be assigned to the specified device. (This does not apply to private sets.)
- System device attribute defines that system files can be assigned to the specified device.

The master device attribute is the only one which can be applied to only one device within the device set. If a file is not a system file and is not meant to be a queue or permanent file, it can be assigned to any device within the device set.

File assignment occurs by first picking the appropriate device within the device set. Appropriateness is partly determined by the various assigned attributes.

The device set and device attributes serve as a means of performing a software configuration of devices irrespective of the hardware configuration of the drives on which the devices are mounted. This distinction is most obvious in the case of removable devices such as 844s (pack is synonymous with device). To complete the description of the total RMS configuration, drive attributes are used. Drive attributes describe the hardware configuration of the drives. The possible attributes include the following.

- RMS type
- Channel numbers
- Equipment number
- Unit number
- Shared equipment
- Shared unit

Correct specification assures the accessibility of drives by the specifying mainframe.

The master device of each device set contains the following disk resident tables.

- Permanent file directory (PFD)
- Permanent file catalog (PFC)
- Set member table (SMT)
- Device allocation map (DAM)
- Logical flaw table (LFT)
- PFC allocation map (PAM)

The size of the tables is controlled by the specification of the NM and NF parameters on the EST macro call for the master device. NF declares the number of permanent files and queue files estimated to reside on the device set, and NM declares the number of devices to be members of the device set. Multiple cycles of a permanent file should be counted as 1 for the NF specification.

NF must be greater than or equal to 1 and less than or equal to 16384. However, if NF is greater than 15872, the number of hash points in PFD will be nonprime, which adversely affects the efficiency of the permanent file hashing algorithm.

Specification of NM affects the disk space allocated to the SMT, DAM and LFT. The number of PRUs reserved for the DAM (disk copy of the RBR) is  $2 * NM$ . Since this number of PRUs is reserved at the time the device is initialized, and installations may want to add additional devices to the device set at a later time, NM should be chosen with a view to expansion. Since generally one PRU is needed for each RBR, for multiple RBR devices NM should be increased by 1 for each two additional RBRs. Special care must be taken to allow for multiple PRU DAM entries. A DAM entry uses two PRUs if the RBR bit table size is 62D CM words. For such DAM entries, NM should be increased by 1 for each additional RBR. The LFT table is the same size as the DAM table. The SMT table is assigned one record block.

The NF parameter affects disk space allocated to the PFD, PFC and PAM. The PFD is allocated  $NF/4$  PRUs (four PFD entries per PRU). The PFC is allocated  $3 * NF/2$  PRUs. A PFC entry always occupies an integer number of PRUs. A PFD entry has a length of 16D words.

The attached permanent file table (APF) contains two-word entries and is central memory resident. Every permanent file in use by a job must have an APF entry. The size of the APF table (L.APF) limits the number of permanent files attached simultaneously by all jobs in the system.

The mounted set table (MST) contains five-word entries and is central-memory resident. Every mounted device set (private and public) is described in the MST. The number of MST entries (N.SETS) limits the number of device sets mounted simultaneously by all jobs in the system.

Each RMS device described in the equipment status table (EST) has an associated DDT entry in the fixed portion of the DDT (four words/entry). The remainder of the DDT is described as the variable portion of the DDT (two words/entry) and is reserved for a list of packs for which jobs are swapped out waiting.

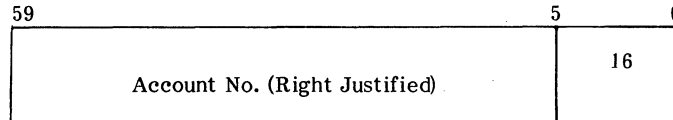
The permanent file routines contain a universal password that, when specified in a request, grants a universal permission. A default file retention period is also defined. This universal password, permission, and file retention period apply to permanent files on private sets created on NOS/BE 1.2 level 461 and earlier systems, and on all public sets. The installation can change the universal password by redefining the symbol UNIV in permanent file PP routines. It is located in COMDECK PREAMB and should be defined as a 9-character value. The installation can change the universal permission by redefining the symbol IP.UP in deck IPARAMS on PL1A. Symbol IP.PFRP in IPARAMS defines the default file retention period. The universal password and file retention period do not apply to private sets created on NOS/BE level 473 and later systems.

A public password defined in the permanent file routines must be specified to permit use of ID=PUBLIC on a CATALOG or RENAME request. This public password applies to private sets created on NOS/BE 1.2 level 461 and earlier systems, and all public sets. The public password can be changed by redefining the symbol IDPERM in permanent file PP routines. It is located in COMDECK PREAMB and should be defined as a 9-character value. This public password does not apply to private sets created on NOS/BE 1.3 level 473 and later systems.



Private sets created on NOS/BE level 473 and later systems have their own universal password, universal permission, public password, and default permanent file retention period stored in the label of the master device. The installation can define default values for the ADDSET parameters UV, UP, PB, and FR by redefining symbol DFV\$ (at UPWDEF.1) in the CP routine PFCCP to be nonzero and by redefining symbols UVD, UPD, PBD, and FRD (at UPWDEF.2 through UPWDEF.5) in PFCCP to contain installation defined default values. When DFV\$ is set to a nonzero value, all four default values must be defined. If the installation does not define any defaults, all private set master device ADDSET statements must contain UV=, UP=, PB=, and FR= parameters. If these parameters are not present, ADDSET aborts with the message xx MUST BE SPECIFIED, where xx is the parameter to be specified.

The installation must provide a PP routine (part of its own accounting routines), to store the user's account number into each control point area, in word W.CPFACT. This account number is presumed to have been taken from the job statement, and is used for CATALOG regardless of any AC parameter specified in the control statement or macro call. The identification, 1 to 9 alphanumeric display-coded characters, has the following format.



The account number should be right-justified to the 16 code (octal) and left-filled with binary zeros. When W.CPFACT is nonzero, accounting dayfile messages are issued to both system and control point dayfiles whenever the status of a permanent file changes; that is, when a catalog, purge, or rename is processed.

### DISK SPACE THRESHOLD SETTING

The unavailability of certain types of disk space can cause deadlocks. These types and the kinds of files that reside on them are listed.

<u>Public Set</u>	<u>Device Attribute</u>	<u>Kind of File</u>
System	PF	Dayfile, CE error file
PF	PF	Default PFs
Q	Q	Default Q files
SCR (can be multiply defined)	--	Default local files, swap/roll files

Two thresholds for each of the four types of disk space are specified as assembly parameters in 2RN. 1RN periodically (every 2\*\*IP.RBINT seconds) initiates the calculation of the sum of available RBs for each of the thresholds. If the available disk space equals or falls below the first or upper threshold, an operator warning message is issued. If the available space falls below the second threshold, the system is placed in step mode and a final operator warning message is issued. The upper threshold for scratch space should be designated high enough (CM size plus 10 percent) to enable the operator to clear control points and initiate one or more jobs to free space.

The system issues only one message for each threshold. When available disk space again exceeds a threshold, a message informs the operator. The installation can disable the feature by defining IP.RBINT=11D.

The interval at which the available RBs are checked against the threshold values is defined in 1RN as an installation assembly parameter (IP.RBINT). The interval in seconds is calculated as 2\*\*IP.RBINT. The default value of IP.RBINT equals two (that is, a four-second interval).

Macro DSTDEF generates the disk space threshold table (DSTT) used by 2RN to calculate the available RBs on the sets defined in the DSTT. The set being checked must have the attributes defined in the macro; however, it may have other attributes as well. This allows installations which have sets with multiple set attributes (for example, PF and Q) which in turn contain devices with a single attribute (for example, a PF device or a Q device) to check the set once for available PF space and once for available Q space with independent thresholds.

On a shared disk, all local space will be released by any mainframes having the device logically OFF in the EST. This may prevent space deadlock situations.

DSTDEF has the following format.

```

DSTDEF      T1,T2,A,B,C,D
T1=         Upper threshold (in RBs)
T2=         Lower threshold (in RBs)
A through D Optional parameters indicating up to four set attributes required on sets to which the thresholds are
              applied
              P   Permanent files
              Q   Queue
              S   System
              X   Scratch
    
```

Calls to the DSTDEF macro should be inserted at DSTDEF.1 in deck 1 RN on PL1A.

Default threshold settings are as follows.

```

DSTDEF      T1=50,T2=20,A=S
DSTDEF      T1=100,T2=50,A=P
DSTDEF      T1=150,T2=100,A=Q
DSTDEF      T1=160,T2=80,A=X
    
```

Threshold examples

Configuration 1

<u>Setname</u>	<u>Set Attributes</u>	<u>VSN</u>	<u>Device Attributes</u>
SYSTEM	SX	844A	MSP
PFQSET	PQ	844B	MPQ
PFQSET	PQ	844C	PQ
PFQSET	PQ	844D	PQ

DSTDEF macro calls

```

DSTDEF      T1=200,T2=100,A=S,B=X
DSTDEF      T1=300,T2=150,A=P,B=Q
    
```

In this example, the operator is warned when the available RBs on VSN 844A reaches 200 or fewer and 100 or fewer, and when the available RBs on VSNs 844B through 844D combined reach 300 or fewer and 150 or fewer.

Configuration 2

<u>Setname</u>	<u>Set Attributes</u>	<u>VSN</u>	<u>Device Attributes</u>
SYSTEM	SX	844W	MSP
PFQSET	PQX	844X	MP
PFQSET	PQX	844Y	Q
PFQSET	PQX	844Z	P

DSTDEF macro calls

```

DSTDEF      T1=250,T2=150,A=X
DSTDEF      T1=100,T2=50,A=S
DSTDEF      T1=200,T2=100,A=Q
DSTDEF      T1=150,T2=50,A=P
    
```

In this example, the operator is warned when the available space for scratch files (VSNs 844W, 844X, 844Y, and 844Z) is 250 and 150 RBs (or fewer). Warnings for PF space occur when VSNs 844X and 844Z contain 150 and 50 available RBs (or fewer), for Q space when VSN 844Y contains 200 and 100 available RBs (or fewer), and for S space when VSN 844W contains 100 and 50 available RBs (or fewer).

## SCHEDULING PARAMETERS

### JOB CLASSES

#### Definitions

- Minimum queue priority (MINQP); the priority with which a job will first enter the CM queue.
- Maximum queue priority (MAXQP); the maximum priority level a job in the CM queue may achieve while waiting for scheduling.
- Base quantum (BQ); the amount of time that a job, once brought to a control point, maintains a high priority, thus helping to avoid being swapped out by another job.
- Quantum priority (QP); the priority given to a job when it has been swapped-in. The job maintains that priority for the duration of its base quantum.
- Age rate (AR); a factor used to weight the priority of a job according to the time spent in the CM queue.

The preceding parameters apply to each of the available classes of jobs. Each class serves to define a series of jobs by their common characteristics, such as response time requirements or the minimum amount of time that a job has access to core.

- Anticipated field length (AFL); an amount of central memory field length which the scheduler tries to set aside in anticipation of jobs of INTERCOM or higher class. Scheduler will not swap in a job from the central memory queue if it is a batch or device class job and if such a swap would not leave at least an amount of unassigned memory equal to AFL.

The seven classes include the following list.

Batch

Device (batch with nonallocatable devices)

INTERCOM

Multuser

Express

Graphics

ECS (batch with direct access ECS)

When a job requests scheduling for central memory, its job descriptor table entry is placed into the central memory queue with a queue priority equal to the minimum queue priority of its class. Its priority is evaluated according to its minimum queue priority, the age rate of the class, time in the queue, and the job statement priority. When the priority of the job reaches the maximum for the class, aging ceases. This priority evaluation is performed for all jobs in the central memory queue, and the results are compared with the priorities of those jobs at control points. When a job is swapped into central memory, it is given a priority equal to the quantum priority of its class.

When the quantum of the job has elapsed, its priority is reduced to the minimum of its class.

Since it requires some overhead to swap a job, the quantum permits a job to remain at a control point for a reasonable length of time before it becomes eligible for swapping. The quantum of a job is considered elapsed when the job has used a specified amount of CPU or PPU time.

All priorities for a class, except MAXQP, are weighted by job statement priority.

Figure II-1-1 illustrates the interaction between two classes, batch and INTERCOM, and displays in a graphical form the relationship between the parameters of these two classes.

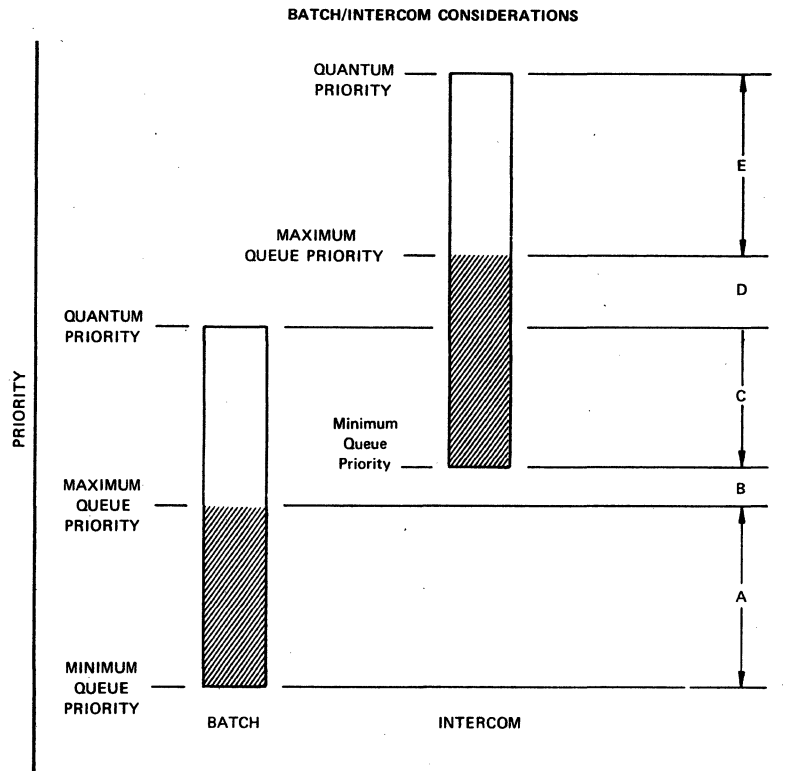


Figure II-1-1. Batch/INTERCOM Considerations

The assumptions used in formulating this set of parameters were that the response time for INTERCOM users should fall within certain bounds, irrespective of the batch loading; that, once a batch job is executing, it has a guaranteed-period of execution before competing with other batch jobs; and that a batch job be allowed to execute a minimum period before a swapout can be forced by an INTERCOM job. Within the batch class, the aging between the minimum queue priority and maximum queue priority (interval A) is intended to ensure that, job statement priority considerations aside, the first batch job to enter the central memory queue will be the first job to be swapped into central memory. The minimum queue priority for an INTERCOM job is greater than the maximum queue priority of the batch job (interval B) so that INTERCOM jobs will not have to compete with batch jobs waiting for central memory. Aging of jobs in the INTERCOM class serves two purposes: first, as in the batch class, to ensure first into the central memory queue, first into central memory; secondly, to allow INTERCOM jobs, after a certain period of time has elapsed, to force the swapout of a batch job so that the INTERCOM job can run.

The extra increment D, between the quantum priority of a batch job and the maximum queue priority of an INTERCOM job, allows INTERCOM jobs to be selective in the batch jobs that are swapped-out to provide core, by becoming eligible to swap, first of all, low job statement priority batch jobs and, eventually, to be able to force out even the highest job statement priority batch jobs. Interval D can be set smaller than the total range of the job statement priority values. By doing so, those jobs with a high job statement priority will not be forced out by INTERCOM jobs before their quantum has expired. Interval E between the maximum queue priority and the quantum priority of INTERCOM jobs, is used for similar purposes as interval C in the relationship between INTERCOM jobs and the next higher class of users. Similarly, this also will allow INTERCOM jobs to run to their quantum before they start to compete for central memory with other INTERCOM jobs.

The following list is the default set of parameters, as they appear in CMR. The parameters selected provide good throughput for an installation running a heavy load of batch jobs, as well as provide good response time for a 20-terminal INTERCOM system where an average of ten terminals are active at any one time. However, graphics jobs, if running, will take precedence over all other jobs.

* SCHEDULER PARAMETER SETTINGS				CMR	1527
* PARAMETER DESCRIPTION				S DISPLAY	CMR 1528
* TYPEIN				CMR	1529
MAXNBA	CEQU	30B	MAX NO OF JOBS W/O NON-ALLOC EQUIP	MAXN1	CMR 1530
MAXNDE	CEQU	10B	MAX NO OF JOBS WITH NON-ALLOC EQUIP	MAXN2	CMR 1531
MAXECC	CEQU	4000B	MAXIMUM ECS COMMITMENT	MAXE	CMR 1532
QPINP	CEQU	2200B	INPUT QUEUE QUANTUM	QP0	CMR 1533
BQINP	CEQU	2000B	INPUT QUEUE BASE QUANTUM	BQ0	CMR 1534
BONECS	CEQU	1	DAECS PRIORITY BONUS	BON	CMR 1535
* BATCH CLASS				CMR	1539
MINQPBA	CEQU	100B	MIN QUEUE PRIORITY	MINQP1	CMR 1540
MAXQPBA	CEQU	1000B	MAX QUEUE PRIORITY	MAXQP1	CMR 1541
ARBA	CEQU	4B	AGING RATE	AR1	CMR 1542
QPBA	CEQU	1400B	QUANTUM PRIORITY	QP1	CMR 1543
BQBA	CEQU	2000B	BASE QUANTUM	BQ1	CMR 1544
* DEVICE CLASS				CMR	1545
MINQPDE	CEQU	200B	MIN QUEUE PRIORITY	MINQP2	CMR 1546
MAXQPDE	CEQU	1000B	MAX QUEUE PRIORITY	MAXQP2	CMR 1547
ARDE	CEQU	10B	AGING RATE	AR2	CMR 1548
QPDE	CEQU	1400B	QUANTUM PRIORITY	QP2	CMR 1549
BQDE	CEQU	2000B	BASE QUANTUM	BQ2	CMR 1550
* INTERCOM CLASS				CMR	1551
MINQPIN	CEQU	1110B	MIN QUEUE PRIORITY	MINQP3	CMR 1552
MAXQPIN	CEQU	2400B	MAX QUEUE PRIORITY	MAXQP3	CMR 1553
ARIN	CEQU	1000B	AGING RATE	AR3	CMR 1554
QPIN	CEQU	2500B	QUANTUM PRIORITY	QP3	CMR 1555
BQIN	CEQU	200B	BASE QUANTUM	BQ3	CMR 1556
* MULTI-USER CLASS				CMR	1557
MINQPMUJ	CEQU	2410B	MIN QUEUE PRIORITY	MINQP4	CMR 1558
MAXQPMUJ	CEQU	2510B	MAX QUEUE PRIORITY	MAXQP4	CMR 1559
ARMUJ	CEQU	200B	AGING RATE	AR4	CMR 1560
QPMUJ	CEQU	3000B	QUANTUM PRIORITY	QP4	CMR 1561
BQMUJ	CEQU	4000B	BASE QUANTUM	BQ4	CMR 1562
* EXPRESS CLASS				CMR	1563
MINQPEXP	CEQU	1000B	MIN QUEUE PRIORITY	MINQP5	CMR 1564
MAXQPEXP	CEQU	3200B	MAX QUEUE PRIORITY	MAXQP5	CMR 1565
AREXP	CEQU	400B	AGING RATE	AR5	CMR 1566
QPEXP	CEQU	3200B	QUANTUM PRIORITY	QP5	CMR 1567
BQEXP	CEQU	400B	BASE QUANTUM	BQ5	CMR 1568
* GRAPHICS CLASS				CMR	1569
MINQPGRA	CEQU	2000B	MIN QUEUE PRIORITY	MINQP6	CMR 1570
MAXQPGRA	CEQU	3500B	MAX QUEUE PRIORITY	MAXQP6	CMR 1571
ARGRA	CEQU	2000B	AGING RATE	AR6	CMR 1572
QPGRA	CEQU	3500B	QUANTUM PRIORITY	QP6	CMR 1573
BQGRA	CEQU	1000B	BASE QUANTUM	BQ6	CMR 1574
* ECS CLASS				CMR	1575
MINQPECS	CEQU	200B	MIN QUEUE PRIORITY	MINQP7	CMR 1576
MAXQPECS	CEQU	770B	MAX QUEUE PRIORITY	MAXQP7	CMR 1577
ARECS	CEQU	4	AGING RATE	AR7	CMR 1578
QPECS	CEQU	2000B	QUANTUM PRIORITY	QP7	CMR 1579
BQECS	CEQU	6000B	BASE QUANTUM	BQ7	CMR 1580
****					CMR 1581

**NOTE**

The values for QPxx, MAXQPxx, and MINQPxx must not exceed 6777B. The system uses priorities greater than 6777B to indicate special conditions.

The foregoing default set of Scheduler parameter settings does not represent the ideal settings for most installations. Adjustments should be made to these parameters so as to match the needs of individual sites. Although the default parameters produce reasonable batch throughput and interactive job response time on a 131K mainframe, the parameter set must be modified for efficient use on a smaller mainframe. The following settings are recommended as a starting point for further tuning systems to individual site requirements.

The 131K parameter set should appear similar to the following:

	IP.MPR	70B
	MAXNBA	6
	MAXNDE	2
	QPINP	2000B
	BQINP	2000B
<u>BATCH CLASS</u>	MINQPBA	0
	MAXQPBA	700B
	ARBA	1
	QPBA	1000B
	BQBA	2000B
<u>DEVICE CLASS</u>	MINQPDE	0
	MAXQPDE	700B
	ARDE	1
	QPDE	1000B
	BQDE	2000B
<u>INTERCOM CLASS</u>	MINQPIN	700B
	MAXQPIN	1420B
	ARIN	100B
	QPIN	1520B
	BQIN	200B
<u>MULTI-USER CLASS</u>	MINQPMUJ	1420B
	MAXQPMUJ	1520B
	ARMUJ	200B
	QPMUJ	1620B
	BQMUJ	7000B
<u>EXPRESS CLASS</u>	MINQPEXP	400B
	MAXQPEXP	1420B
	AREXP	200B
	QPEXP	1420B
	BQEXP	4000B
<u>GRAPHICS CLASS</u>	MINQPGRA	1420B
	MAXQPGRA	1530B
	ARGRA	200B
	QPGRA	1630B
	BQGRA	2000B
<u>ECS CLASS</u>	MINQPECS	0
	MAXQPECS	670B
	ARECS	1
	QPECS	1400B
	BQECS	6000B

The parameter set for a 65K mainframe should be identical with the exception of MAXNBA dropping to a value of 2.

The batch classes have low minimum and maximum queue priorities as well as low age rates. The device class with twice the age rate of the batch class gives the device class a scheduling advantage over the batch class. A device class job would experience, on the average, half the wait time of a batch class job. Since the device class represents additional resources being tied up, such as control points and tapes, it is preferable to get that job through the system with a minimal delay.

The quantum priorities of the batch and device classes are low enough so that INTERCOM jobs, having a high age rate, can force batch jobs to be swapped out after a one-half to two second delay, depending on job card priority and quantum considerations.

The INTERCOM class job is given a small base quantum which normally is enough time to execute an INTERCOM job step. The batch quantum, on the other hand, is larger, preventing batch jobs from swapping other batch jobs unnecessarily. The multiuser class job, such as EDITOR, is given the highest priority because it can service several INTERCOM users simultaneously.

The parameter MAXN determines the maximum number of batch or device class jobs which can run at any given time. The number of device class jobs is kept small; the determining factor being that device class jobs are rolled out rather than swapped out; each job can make a control point unavailable for swapping. It is essential to keep a reasonable number of control points available for serving other jobs. The maximum number of batch jobs is much higher, a large number being preferable to provide the Scheduler with a better pool of job candidates, allowing better core utilization. However, too large a job pool may adversely affect individual job turn-around while improving total system throughput.

Parameters QP0 and BQ0 in the lower half of the S display are the quantum priority and base quantum given to jobs coming out of the input queue and entering a control point for the first time. The quantum priority is higher than that for normal batch jobs, enabling short jobs to run to completion without swapping.

The express queue is given a high priority and aging rate, since it contains all jobs terminated by operator intervention. The quantum is small because the end-of-job procedure is minimal. This class was given express consideration under the assumption that these jobs would release valuable resources back to the system.

## CENTRAL MEMORY

The determination as to which jobs may occupy central memory simultaneously depends on the job class, job field length, and the available central memory field length. Additional considerations affect central memory allocations such as maximum field length (MFL) and anticipated field length (AFL).

- |     |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       |
|-----|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| MFL | This value represents the largest amount of central memory field length any single job may obtain. MFL is set by installation parameter IP.MFL but may be changed by the protected DSD S-display command, MFL,nnnn, which changes MFL to nnnn*100B CM words. (nnnn must not exceed 3777B.)                                                                                                                                                                                                                                                                                                            |
| AFL | This value is the CM FL which Scheduler will set aside for jobs of INTERCOM or higher classes. Only batch or device class jobs waiting in the CM queue are affected by this parameter as the value of AFL is subtracted from the field length available to these two job classes before determining if sufficient field length is available for the job. AFL is initially set to AFL.BAS*1000B CM words. When INTERCOM is up, AFL is incremented by AFL.INT*1000B CM words. As with MFL, AFL may be changed by the protected DSD S-display command, AFL,nnn, which changes AFL to nnn*1000B CM words. |

AFL is used to minimize the effect of the following events.

When an INTERCOM job step ends and the control point and field length are freed, Scheduler assigns this control point and field length to a batch job from the CM queue. Another INTERCOM job may displace this or another batch job because INTERCOM jobs have a higher priority. When this happens frequently, the repetition of this cycle can become a real problem.

To reduce the frequency of this swapping, Scheduler requires that a certain amount of CM remain unassigned and thus available for initiation of an INTERCOM job step, but not available for swap-in consideration to batch or device class jobs. This amount of CM is called the anticipated field length (AFL). The value AFL is defined by two symbols in CMR.

AFL.BAS Basic AFL is used if INTERCOM is not up.

AFL.INT INTERCOM increment is the amount added to basic AFL when INTERCOM is brought up and subtracted when INTERCOM is dropped.

The following guidelines should be used to arrive at a practical setting for MFL and AFL for each individual installation.

- |       |                                                                                                                   |
|-------|-------------------------------------------------------------------------------------------------------------------|
| PFL   | Physical Field Length, in CM words                                                                                |
| CMR   | Size of CMR (LWA of library)                                                                                      |
| RBT   | Size of RBT area                                                                                                  |
| POSFL | 2 * IP.POSFL * 100B                                                                                               |
| FFL   | Fixed field length, total field length in use by system routines (JANUS, INTERCOM, SCPs, 7000 Station, and so on) |
| UFL   | User field length, available for user jobs                                                                        |
| MUFL  | Maximum UFL (equal to UFL when FFL=0)                                                                             |
|       | MUFL = PFL - CMR - RBT - POSFL                                                                                    |

AFL.INT should be set to a value which approximates the average FL occupied by INTERCOM jobs at one time.

If the installation experiences a large number of swapouts caused by requests for increased FL by jobs at control points, AFL.BAS should be increased to counteract this effect.

MFL may be set to any appropriate value within the following limitations.

- Too small a value may preclude some standard software products from executing.
- If MFL is set greater than MUFL, a job requesting this much FL will be swapped out and cannot be swapped back in to complete.

If MUFL > MFL > UFL, a job requesting a field length of MFL will be swapped out and cannot be swapped back in to complete until at least one of the system routines is terminated. This may occur if a system routine is initiated during the time that a very large job is executing. This condition may effectively remove nonallocatable resources such as tape drives or permanent files from the system when, for example, INTERCOM is initiated. Console operators should be advised that this problem may occur.

If at any time MFL>UFL-AFL, a job in batch or device class which has (or is requesting) a field length of MFL may be swapped out and Scheduler will not normally swap it back in. The job may be allowed to swap back in by reducing AFL or by dropping INTERCOM, or by locking the job in with the N.LOCKIN. command. Another alternative is to drop a system routine.

It is not recommended that MFL be changed dynamically while the system is processing jobs. It should be changed only during system initialization to correct an incorrect setting of the parameter when the system was built. To do otherwise could cause certain jobs to abort when they request additional memory.

## ECS USAGE

ECS usage is affected by several factors that can be changed by console entries.

MFLE	The largest amount of ECS that any user can obtain. This value is established by IP.MECS. Refer to the NOS/BE Operator's Guide for a description of the MFLE S-display command.
MAXECC	The total amount of direct access ECS committed to active jobs. For efficient use of ECS, this value should normally be set higher than the actual amount of available ECS. Too high a value will cause excessive scheduling and swapping. Refer to the NOS/BE Operator's Guide for a description of the MAXE S-display command.
BONECS	The amount of extra priority given every job with ECS assigned. System efficiency improves if the swapping of user field lengths can be minimized. Refer to the NOS/BE Operator's Guide for a description of the BON S-display command.

## CPU

CPMTR selects which job the CPU should be assigned to next. Using the parameter values with which the system is released, this selection occurs at least every 20 milliseconds for each CPU.

The selection is based on the CPU priority level associated with each job. As released, there are five possible levels. They are defined by the following symbols which are defined in IPARAMS.

PR.IDLE (0)	A low priority job for default assignment
PR.BATCH (1)	All normal batch jobs execute at this priority.
PR.INT (PR.BATCH+1)	INTERCOM job
PR.SCP (PR.INT+1)	System control points and jobs initiated by the operator
PR.SYS (PR.SCP+1)	System jobs (storage move and scheduler)

Each of these levels may be redefined by inserting a new value into IPARAMS in the same manner that IP.xx symbols are redefined. Note that PR.INT is defined relative to PR.BATCH, and so forth, so that when any one level is redefined, all higher levels are automatically redefined. PR.SYS must always be the highest priority level.

Because INTERCOM jobs have a higher CPU priority, it is possible for an errant INTERCOM job to seriously degrade the batch jobs. Installations that want to avoid this possibility should insert the statement PR.INT CEQU 1 into IPARAMS. This leaves INTERCOM jobs running on an equal basis with BATCH jobs.

## SYSTEM IDLE MODE

If IP.SIDLE is nonzero, code is assembled to support an IDLE mode of operation. When the system is in IDLE mode, control point activity is inhibited; the CPU is not scheduled to any jobs at control points and no jobs are initiated or swapped into vacant control points. If time permits, IDLE mode swaps out all control points and performs a system level checkpoint. Such a checkpoint may be used during deadstart to recover the system even if memory contents are subsequently lost.

The system uses IDLE mode to control activity and possibly improve system recoverability in the event of circumstances which jeopardize system availability. IDLE mode may also be initiated by operator command; in this case, a checkpoint is always performed. This feature may be useful in idling system activity prior to a scheduled system downtime (for example, prior to a preventive maintenance period).



**NOTE**

IDLE mode does not provide a completely restartable system checkpoint. Jobs which cannot be swapped out are not recovered but are either rerun or dropped if they have no-rerun status. Included in this class are jobs with nonallocatable equipment assigned, jobs waiting for operator action, real-time jobs, and jobs with direct access ECS assigned if swapping of direct access ECS is disabled. Furthermore, jobs swapped out to ECS when IDLE mode is initiated are rerun/dropped upon checkpoint recovery.

**ECS INSTALLATION PARAMETERS (ECSCOM)**

The ECS extensions are designed primarily to improve the efficiency of an I/O bound system by accomplishing the following.

- Buffering the sequentially accessed RMS files through ECS
- Swapping jobs to ECS
- Moving a part of the system library to ECS
- Allocating files in ECS
- Moving part of CMR code to ECS to free up CM space

The default values of the ECSCOM configuration parameters are defined with the CEQU or CMICRO macros, so that an installation can insert all modifications at one given place. The CEQU and CMICRO macros are used to define variables conditionally. Since they are effective only if the variables have not been previously defined, any modifications should precede the default definitions.

Installation parameters oriented to ECS are defined in the COMDECK.ECSCOM. Changes may be made at ECSCOM.8 in an update of PL1A. Default values and parameter descriptions are as follows.

**IP.EBUF (16D)**

Defines the default ECS buffer size in pages. To significantly improve system I/O, the ECS buffer allocated to a file should be at least four times larger than the buffer used in CM for the same file, resulting in a default value in the 10000 to 20000 (octal) words range. A larger ECS buffer (40K or more) does not provide any significant improvement compared with the default value.

If an ECS buffered file does not overflow its buffer, it stays in ECS and is processed as an ECS resident file, possibly locking a very large amount of ECS for only one file. Buffer space is not reserved when the buffer is requested; it is allocated only when needed and released as soon as possible, one page at a time. Allocation of an ECS buffer to a file having a CM buffer approximating one RB does not improve throughput because of the scheduling algorithm used by the stack processor.

If 819 buffering is enabled (IP.819 is nonzero), the parameter IP.EBUF has no effect for 819 buffering. The default buffer size is not variable.

**IP.ELIB (0)**

If zero, the code for ECS resident library will not be assembled in the system; if nonzero, maximum number of words/1000 (octal) that may be used for storing ECS resident library programs. This value can be changed at deadstart time; however, it can be nonzero only if IP.ECSB is nonzero. IP.ELIB cannot be set equal to the paged ECS area total length, since part of it is issued for the system area including empty page stack, system subpages (at least 2K), preallocation page reservation table (1K), and ECS system buffers.

**IP.ERES (0)**

If set to one, the ECS resident file capability is activated. The ECS resident file option feature can improve system throughput for a given job by keeping large files (particularly random access) in ECS. However, it can have an adverse effect on the overall improvement of the system by drastically reducing the amount of ECS available for job swapping, ECS buffering, and the system library.

IP.ECSW (0)

If IP.ECSW is zero, the code that allows swapping of user direct access ECS is disabled. If not zero, batch jobs with ECS requirements (but with no nonallocatable device requirements) will be assigned to the ECS class, and will be scheduled in the same way as batch class jobs. Note that more resources will be used to swap ECS field lengths to mass storage, but system throughput will improve. If IP.ECSW is nonzero, IP.ECSB must also be nonzero.

SYSTEM CIRCULAR BUFFERS (SCB)

Data is buffered between ECS I/O buffers and disks by the system circular buffers. There are two types of SCB: CM and DDP. In both cases, the data transfer is controlled by the 1SP I/O buffering executive and circular buffer manager (CBM), a CP monitor function. CBM is activated by 1SP to start and to end the transfer and by MTR during the transfer through a system M.BUFFER function. The transfer is controlled by the SCB control table which has FIRST, IN, OUT, LIMIT, and TRIGGER pointers. The CM circular buffer that the control table points to is 101B words/PRU in length for a CM SCB and 1B word/PRU in length for a DDP SCB. Minimum size for a SCB is 3 PRUs. Larger SCBs cause fewer CPU interrupts, but require more ECS to be allocated for a read ahead.

SCBs are defined by the ECSBUF macro as follows.

ECSBUF	size,DDP
size	PRU count; SCB size
DDP	blank; CM SCB
	DDP; DDP SCB

There must be one ECSBUF macro call for each SCB defined. In addition, each DDP SCB must have a DDP defined in EST. If more DDPs are defined in the EST than there are DDP SCBs, the additional DDPs are not used. If fewer DDPs are defined in the EST than there are DDP SCBs, the additional SCBs are not used. The number of CM SCBs should be limited to three. The number of DDP SCBs corresponds with the number of available DDP channels. Also, the number of SCBs must be the same as the number of RMS controllers.

An example of SCB installation changes follows.

Assuming that ECS buffering is defined (IP.ECSB = 1), there is one DDP available and if two CM SCBs are to be defined, the installation changes are as follows.

```

*INSERT IPARAMS.15
(other installation parameter modifications)
IP.ECSB CEQU
*INSERT EST.1
(other EST modifications)
ED EST CH=12,EQP=5
*INSERT ECSBUF.1
ECSBUF 24,DDP
ECSBUF 16
ECSBUF

```

Data can be transferred between ECS and RMS using either type of SCM interchangeably. The priority of the SCBs, as determined by the order of the ECSBUF macros, determines which SCB is assigned when one is requested. The first SCB that is not busy is the one assigned. In the previous configuration, three SCBs are defined with DDP having the highest priority.

At least one SCB is required if ECS swapping (IP.ECSW) is enabled.

MACROS TO CONSTRUCT ECS LABEL

Before ECS is used for the system,† it is divided into partitions and a label that defines these partitions is written to ECS. The image of an ECS label can be set up in CMR for deadstart processing, so that the operator can construct an ECS label when required.

An ECS label consists of a two-word header and two-word entries for each partition. The number of partitions is limited to nine (one for a COMMON partition and two per mainframe for as many as four mainframes).

The ECS label header contains the length of defined ECS and the number of defined partitions with a specially formatted header word and a checksum. The ECS label is written in the first 1000B words of ECS to one of the areas starting at 120B, 230B, 340B, 450B, 560B, or 670B.

†When constructing the ECS label, ECS on a model 176 cannot be shared between mainframes; all partitions of ECS must be associated with the 176.

Two macros are available to set up a temporary image of an ECS label in CMR at location 340B. This information will be initialized during deadstart continuation.

ECSLABEL	length
length	Defined ECS length/1000B
ECSPART	name,type,fl,bit,fw
name	Partition name (up to 10 characters), which is the mainframe ID unless the partition is type COMMON.
type	Partition type: <ol style="list-style-type: none"><li>1 Direct access ECS area. One partition of this type must be assigned to each mainframe ID. This area contains the ECS system segments.</li><li>2 System area and paged ECS area. One partition of this type must be assigned to each mainframe ID. This area includes ECS system tables and buffers and may contain ECS resident libraries and files.</li><li>3 COMMON area for multicomputers. First partition defined in ECS must be type COMMON if ECS link is defined.</li></ol>
fl	Partition length/1000B.
bit	Reserved for future use.
fw	First word address/1000B of partition. If absent, LWA+1/1000B of preceding partition is assumed.



The following example assumes the size of ECS is 754000B (250K decimal) words and divides ECS for two computers and a COMMON partition.

```
ECSLABEL      754
ECSPART       LINK,3,10
ECSPART       ONEOVEM,1,40
ECSPART       THEOTHER,1,40
ECSPART       ONEOVEM,2,300
ECSPART       THEOTHER,2,300
```

43000B words of ECS remain unassigned in the example.

The definition of ECS label can be placed at CMR.2167.

The computer identification label for a mainframe may optionally be stored in CMR. When partitions are assigned at deadstart, the mainframe ID from CMR is compared against ID of the partitions. The operator can modify the mainframe ID from the console during deadstart ECS partitioning processing.

A computer ID is stored into CMR by defining the configuration parameter for computer ID, IP.CMPID. This definition, if changed, should be placed at CMRIP.1.

```
Example      IP.CMPID      CMICRO      10,(SN58)
```

### ECS SYSTEM SEGMENTS

The segments in an ECS system have all their nonlocal symbols defined in text CMRTEXT. If none of those symbols are affected, the segments need not be reassembled when configuring a system. Deck CMRDIR defines the residence of the segments. A segment must be named in an OVL macro call.

The format of the call follows.

```
OVL      segment, CM, iparam

segment  The segment name

CM       Indicates segment should be resident, no matter where it is defined.

iparam   Installation parameter when specified. The segment is only defined if the installation parameter is nonzero.
```

OVL calls follow a call to the AREA macro, defining the overlay area where the corresponding segments execute.

AREA name

Three areas are defined for use as name.

```
CM       Central memory resident
MTR      Monitor mode overlay area
USER     User mode overlay area
```

Segments may be moved to CM residence by defining them within the CM resident area, or by adding the CM parameter to the corresponding OVL call. No other change of area is allowed.

CMRDIR as released contains a complete template for the system. All segments defined in the CM area are resident segments and cannot be moved to another area. Changes should not be made to CMRDIR unless they result in significant performance improvement. Segments not defined in CMRDIR are not included in the ECS system built by LDCMR.

### SEGMENT ACTIVITY COUNT

A provision has been made to count how many times each ECS resident segment is being referenced to aid the installation in determining how many segments should be moved into central memory. This count may be activated via an SAC call in RA+1. The interface description follows. The installation can write a user program to request the data and format it into a usable report.

### CALLING SEQUENCE

```
label     SYSTEM,SAC,R,addr
```

R is required recall parameter.

The area specified by addr is at least twice the number of segments in length.

The code for counting segment calls replaces the code that performs the segment trace. The counts are accumulated in the segment trace buffer. On each call, the counts are transferred from the trace buffer to the user's buffer at addr, and the trace buffer is reset to zeros.

The first time that the segment activity count is called, the trace buffer contains trace data instead of segment activity counts. In this case, the data that is returned should be ignored.

The format of the data returned at addr follows.

addr contains the length of the data that was returned. addr/2 is the number of segments.

addr+1 is zero.

addr+2n is the name of segment number n, left-justified, zero-filled.

addr+2n+1 is the number of segment calls since the last call for segment activity counts.

The DSD command, RESTART, causes tracing to be resumed.

If ADDR+2n is not within the field length of the calling program, the job is aborted with the message PP CALL ERROR.

If the trace buffer is smaller than n words, the segment activity count code is not placed into CMR to initiate activity counting.

#### ECS AS A LINK MEDIUM BETWEEN MAINFRAMES

ECS can be used as a link medium device† between two mainframes (CX) as an alternative to 6683/6683 channel couplers (CC). To assemble the system with the ECS driver turned on, set IP.ECSLK nonzero. IP.ECSB must also be nonzero. The following ECSCOM installation parameters can be varied by the installation for the ECS driver.

IP.ECSLK	CEQU	0	0 = no MMF ECS link, 1 = ECS link can be used.
IP.LNKBF	CEQU	1	Number of MMF ECS link buffers defined. One link buffer is required for each link between two distinct mainframes.
IP.MAXBL	CEQU	1461B	Size of maximum block to be transferred on one ECS access. This cannot exceed 1461B.
IP.LNKMN	CEQU	500B	Minimum acceptable ECS buffer length.
IP.ECYC	CEQU	37B	Controls primary restart cycle of the ECS link driver. Driver restarts at primary cycle rate only if there is work to do. The following restart times are approximate in milliseconds.
		3B	1 millisecond
		7B	2 milliseconds
		17B	4 milliseconds
		37B	8 milliseconds
		77B	16 milliseconds
		177B	32 milliseconds
		377B	128 milliseconds
		1777B	256 milliseconds
		3777B	512 milliseconds
		7777B	1024 milliseconds
IP.CYSTP	CEQU	1	Controls the rate at which the link driver slows its restart rate when the link activity is low.
IP.EIDLE	CEQU	5	Controls the number of idle cycles allowed before the link decides to slow the restart rate due to inactivity.
IP.ECLNK	CEQU	0	0 = no simulation for dual test mode, 1 = allows simulation of dual computers for ECS link testing on one mainframe. IP.LNKBF must be set to 2.

† On a CDC CYBER 170 Model 176, ECS cannot be used as a link medium between two mainframes.

## UPDATE INSTALLATION OPTIONS

The following UPDATE features are available or unavailable through assembly options and may be modified by deleting the appropriate entry in the range UPDATE.703 through UPDATE.711; these changes should be specified in the installation deck PLICI.

DECLKEY	Enables DECLARE directive.
CHAR64	Supports full 64-character set.
PMODKEY	Enables PULLMOD card and G option.
AUDITKEY	Allows audit functions.
EDITKEY	Allows merge and edit.
EXTOVLP	Enables detection of four types of overlap involving two or more cards in a correction set.
DYNAMFL	Declares dynamic table expansion. When this option is assembled, UPDATE automatically expands tables as required and dynamically requests the system 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.

An attempt to use features when the option has not been assembled causes UPDATE to issue error messages. For example, when PMODKEY is not set, the PULLMOD card is not recognized as a legal directive.

All of the above features are enabled by default.

## COMMON MEMORY MANAGER VERSION 1 (CMM)

CMM provides control over all dynamic memory in the field length of a job. Its features are described in the Common Memory Manager Reference Manual. Products that use the CMM include the following.

- CDC CYBER Loader 1
- COBOL 4
- COBOL 5
- Sort/Merge 4
- CDCS 1
- CDCS 2
- QUERY UPDATE 3
- FORM 1
- FORTRAN Common Library 4
- FORTRAN Common Library 5
- BASIC 3
- COBOL Conversion Aids 4
- Data Base Utilities 1
- CDC CYBER Record Manager (BAM and AAM)
- PL/I 1
- SYMPL 1
- FORTRAN Data Base Facility

CMM requires the same minimum hardware configuration as the operating system.

CMM uses symbol definitions from common deck CMMCOM. IPARAMS symbols, which specify the operating system, are also referenced. The following CMMCOM installation parameters can be changed by the installation for CMM.

<u>Name</u>	<u>Default</u>	<u>Description</u>
DEFVER	0	Defines which CMM version is to be used by default.  0 A nonerror checking version (referred to as FAST) is used. 1 An error checking version (referred to as SAFE) is used.
FLF	2000B	If 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.

## DEADSTART LOADING THE OPERATING SYSTEM

The operating system must be loaded before the computer can execute jobs. This procedure involves operator action depending on the type of magnetic tape unit available. The operator should be aware that when the deadstart button is pressed too long, multiple deadstarts might occur that can cause the deadstart tape to be read prematurely, possibly overwriting critical information. To avoid such a possibility, ensure that the deadstart tape is at the load point, activate the deadstart button or switch, and then press the ready button on the tape unit. Refer to the NOS/BE Operator's Guide for the correct procedure for warmstarting and coldstarting the system.

## CONTROLWARE BINARY CREATION

The 66x coldstart card deck can be recreated by running the following job.

```
Job Statement.  
REQUEST(OLDPL,E,HY) NOS/BE 1 PL1A  
REWIND(OLDPL)  
UPDATE(Q)  
COMPASS(I=COMPILE,S=PPTEXT,L=0,B=PUNCHB)  
7/8/9
```

```
*COMPILE ABC  
6/7/8/9
```

This assembly will produce a small coldstart binary card deck. The coldstart card deck including 66x controlware will be constructed as follows:

```
BINARY CARDS OF ABC  
7/8/9  
66x CONTROLWARE DECK (unprefixed, that is, deck usable as input to COPBC)  
6/7/8/9
```

The 66x, 844, and 885 type controlware binary card decks must be run through the following job to obtain card decks of OMT for 66x controllers, OSY for 7054 controllers, OSZ for 7154 controllers, and OSJ for 7155 controllers. Each controlware binary must be run on a separate job. OMT, OSY, OSZ, and OSJ are the binary decks to be used by the DST1 and DST2 build jobs. (Note that these binaries contain 77 tables.)

```
Job Statement.  
COPYBR(INPUT,TAPE1)  
COPYBF(DUMMY,TAPE1)  
REWIND(TAPE1)  
COPBC.  
REWIND(BIN)  
COPYBF(BIN,PUNCHB)  
7/8/9  
Respective controlware binary  
7/8/9  
Additional code for respective device types  
6/7/8/9
```

The additional code should include the following for respective device types.

### 66x Controller

```
01 77000016000000000000  
01 33152400000000000000  
15 00000000000000000000  
01 33152400000000000000
```

### 7054 Controller

```
01 77000016000000000000  
01 33233100000000000000  
15 00000000000000000000  
01 33233100000000000000
```



7154 Controller

01 77000016000000000000  
01 33233200000000000000  
15 00000000000000000000  
01 33233200000000000000

7155 Controller

01 77000016000000000000  
01 33231200000000000000  
15 00000000000000000000  
01 33231200000000000000

COPBC is a system program that converts a controlware binary deck into SCOPE binary format with a prefix (7700) table.

To make the 7152 controlware for 844 full track mass storage and 667/669 magnetic tape usable for input to a deadstart creation job, create the following file using EDITOR under INTERCOM.

```
Job statement.  
REQUEST(MTSTAPE,MT,HY,NORING)  
COPYBF(MTSTAPE,JUNK)    Position tape to correct record  
COPYBR(MTSTAPE,TAPE1)  
COPYBF(DUMMY,TAPE1)  
REWIND,(TAPE1)  
REQUEST(BIN,*PF)  
COPBC.  
REWIND(BIN)  
CATALOG(BIN,CONTWR,ID=CCT)  
*EOR  
  Additional code for respective device types (shown previously)  
*EOF
```

The file can be saved and submitted to the input queue.

SAMPLE JOB FOR CREATION OF INSTALLATION CMRs and CMR LIBRARIES

```

CMRS,IO0,T0,MT1.
COMMENT. CONTROL STATEMENT SEQUENCE FOR 819 SUBSYSTEM
LABEL(PL1 G,D=HY,L=PL1 G,NORING,R)          (819 subsystem only)
UPDATE(Q,P=PL1G,C=C819)                    (819 subsystem only)
UNLOAD(PL1 G)                               (819 subsystem only)
LABEL(PL1 A,D=HY,L=PL1 A,NORING,R)
COMMENT. ASSEMBLE PPTXT
UPDATE(Q,P=PL1 A,C=PPTXT)
COMPASS(I=PPTXT,L=0,S=0,B=PPTXT)
COMMENT. START REPEATABLE CONTROL STATEMENT SEQUENCE, WHERE
COMMENT. EACH REPETITION ASSEMBLES A NEW CMR.
UPDATE(Q,P=PL1 A)
RFL(75000)
COMPASS(I,S=0,B=CMRTEXT,L=0)                CMRTEXT          (ECS system only)
COMPASS(I,G=PPTXT,L=0)                     CMR
COMPASS(I,G=PPTXT,G=CMRTEXT,L=0,B=CMRLIB)  CMR SEGMENTS      (ECS system only)
COMPASS(I=C819,S=PPTXT,G=CMRTEXT,L=0,B=CMRLIB) 819 SEGMENT      (819 subsystem only)
REWIND(CMRTEXT)
COPYBF(CMRTEXT,CMRLIB)
CATALOG(CMRLIB,CMRL108,ID=CCT,XR=XYZ,PW=XYZ,RP=20)
RETURN(CMRTEXT,CMRLIB)
COMMENT. END OF CONTROL STATEMENT SEQUENCE FOR CMRL108.
COMMENT. REPEAT ABOVE CONTROL STATEMENT SEQUENCE FOR ADDITIONAL CMRS
COMMENT. ADDING INPUT RECORDS AS APPROPRIATE.
COMMENT. 64 CMRS = MAXIMUM.
REWIND(LGO)
REQUEST(CMR,*PF)
REWIND(CMR)
COPYBF(LGO,CMR)
CATALOG(CMR,ID=CCT,XR=XYZ,PW=XYZ,RP=20)
7/8/9                                       (819 subsystem only)
*C LCM                                       (819 subsystem only)
7/8/9
*C PPTXT
7/8/9
*/ INPUT RECORD
*I CMRL108
*I IPARAMS.15
*/ IPARAMS MODIFICATIONS FOLLOW THIS STATEMENT
*/
*I CMRIP.1
*/ CMR INSTALLATION PARAMETER MODIFICATIONS FOLLOW THIS STATEMENT
*/
*I CMR.1
*/ CMR MODIFICATIONS FOLLOW THIS STATEMENT
*/
*I LID.1
*/ LOGICAL ID TABLE MODIFICATIONS FOLLOW THIS STATEMENT
*/
*I EST.1
*/ EST CONFIGURATION FOLLOWS THIS STATEMENT
*/
*I RBR.1
*/ RBR ENTRIES FOLLOW THIS STATEMENT
*/
*I FLAW.1
*/ FLAW ENTRIES FOLLOW THIS STATEMENT
*/
*I MUX.1
*/ MUX ENTRIES FOLLOW THIS STATEMENT
*/
*COMPILE    CMRTEXT,CWEOR1                (ECS system only)
*COMPILE    CMR
*COMPILE    CWEOR2.CWEOR3                (ECS system only)
7/8/9
6/7/8/9

```

The following constraints apply to building a multiple CMR deadstart tape. If a CMR has a separate ECS library, the name of that library must be the same as that defined by IP.SLIB (CMRIP.1) in the corresponding CMR. Systems without ECS do not need CMR segments and should disregard all statements applicable to ECS and the 819 subsystem in the preceding example. CDC CYBER 170 Model 176 systems with the 819 subsystem hardware require all statements in the preceding example.

The deadstart tape generation job DST1 (which needs ECS defined) should be examined to realize how CMR handling is accommodated. DST1 may then be modified appropriately to capture the files created by job CMRS in the preceding example. For subsequent running system captures, jobs DST2 and DST3 can be modified to include additional CMR libraries.

#### SYSTEM RESIDENCY

The unconfigured deadstart tape contains 8DN, 8XS, 8XT, 8X8, 3DO, 4DO, A, 1SQ, AAA, 1SP, and its 3SZ system device overlays as CM resident. These routines must be declared CM resident when a deadstart tape is created. Additional routines such as CIO, 4ES, and 1AJ may be made CM resident to enhance system throughput. Routines established as CM resident in the running system will have CM residency on the new deadstart tape created by jobs DST2 and/or DST3.

Note that ECS residency cannot be carried on a deadstart tape. Routines moved to ECS resident will be established as disk resident on a new deadstart tape. The only way to set ECS residency is by MOVE directive changes to the running system. None of the mentioned routines can be moved safely to ECS.

System ECS resident routines occupy a part of the paged area. The paged area is defined in terms of a page stack and accessed through a CMR central processor program (CP.CIO).

ECS residency cannot be specified via EDITLIB creation of the deadstart tape. An EDITLIB run must be performed after deadstart to move routines from their residence as loaded by deadstart to ECS.

When EDITLIB creates a deadstart tape, it terminates the tape with a double end-of-file, effectively creating a null file following the last system library file on the tape. If this last file is not null during the preloading process (after system library files have been copied to mass storage), a job of the following structure is assumed.

```
Job Statement.  
EDITLIB(SYSTEM)  
7/8/9  
EDITLIB Directives  
6/7/8/9
```

This job will be copied to mass storage and cataloged as a permanent file with the following parameters.

```
LFN = ZZZZCS  
PFN = ZZZZCS  
ID = SYSTEM (granted automatically for control point 0 permanent file operations)  
TK = SYSECSLIB  
XR = ECSLIB
```

This job will be run automatically by the terminate deadstart sequence PP program (TDS) whenever ECS is up and the deadstart level is either 0 (preload from tape) or 1 (load from the system permanent file).

Because of restrictions imposed for system (control point 0) permanent file operations, a user cannot catalog a new file with an ID of SYSTEM. Thus, ZZZZCS can be created only in the manner just described. Thereafter, the job can be modified by new-cycle catalog and old-cycle purges with the appropriate permissions (ID=SYSTEM allowed and required).

#### ENHANCED STATION SUPPORT; SPOTS (SPUN OFF TASKS) CORE REQUIREMENTS

The station control point MFSTAT handles all communication between the station and SCOPE 2; or between two CDC CYBER 170 or 6000 Series or CDC CYBER 70 Model 71, 72, 73 or 74 mainframes. In the following description of the station, the term 6000 refers to a CDC CYBER 170 Series Model 171, 172, 173, 174, 175, 720, 730, 750, or 760, CDC CYBER 70 Model 71, 72, 73 or 74, or 6000 Series mainframe. The term 7000 refers to a CDC CYBER 70 Model 76 or 7000 Series mainframe. A spun-off task is a job that MFSTAT initiates and places into the host mainframe input queue after MFSTAT detects a request for an I/O transfer or a staging operation. There are five unique spun-off tasks, permanent file SPOT, tape staging SPOT, spooling SPOT, dump SPOT, and deadstart SPOT.

The spooling SPOT is the only SPOT that transfers more than one file. The spooling SPOT is initiated when communication is established between mainframes and will not terminate until the station is dropped. The two SPOTs activated for spooling are SOT66 which transfers data between two 6000s, and SOT76 which transfers data between a 7000 and a 6000. All other SPOTs are initiated as required to transfer one file. Each SPOT is terminated after completion.

Field length requirements fluctuate as file transfers are initiated and terminated. As files are terminated, buffers are released. The lengths of the buffers used by the spun-off tasks are controlled by definitions in the common deck COMTUNE. Each spot uses one of the DEFs to determine the length of the I/O buffers. All spun-off tasks are written in the SYMPL language.

The nominal release definition values for the I/O buffers are the following.

- LBUFDSPF = 1540=3004B (deadstart via permanent file)
- LBUFDSTP = 1540=3004B (deadstart via tape)
- LBUFDSLM = 4053=7725B (deadstart link buffer)
- LBUFDP = 2049=4001B (7000 dump)
- LBUFLM = 769=1401B (link medium)
- LBUFPF76 = 2601=5051B (permanent file for 6000-7000)
- LBUFPF66 = 2081=4041B (permanent file for 6000-6000)
- LBUFSP = 1041=2021B (spooling)
- LBUFTP = 3079=6007B (tape staging)

Buffers required to transfer a file vary among the SPOTs. All SPOTs transfer a file between a 6000 disk and a link medium device. For some SPOTs, this requires two separate buffers as data is converted. Other SPOTs use only one buffer. When two buffers are needed, the length of the link medium buffer is defined by LBUFLM, and the buffer for the disk is defined by the appropriate symbol (for example, LBUFSP for the spooling SPOT).

Core requirements for various spots are shown in the following chart.

<u>SPOT Type</u>	<u>Code</u>	<u>Buffer Lengths Used to Transfer Files</u>	<u>Total Core Used†</u>	<u>Notes</u>
<b>Spooling</b>				
6000-6000	3700B	LBUFSP+LBUFLM	13200B	Single file transfer
7000-6000	5100B	LBUFSP+LBUFLM	14600B	Single file transfer
<b>Permanent File</b>				
6000-6000	1600B	LBUFPF66	5700B	
7000-6000	2300B	LBUFPF76+LBUFLM	11000B	
<b>Tape Staging</b>				
7000-6000	3700B	3* MBL + LBUFLM/2+3	7600B	Maximum block length (MBL) as specified by user in tape staging. If $MBL < (LBUFTP-LBUFLM/2-3)/3$ [7690]
		LBUFTP	11700B	If $(LBUFTP-LBUFLM/2-3)/3$ [7690] $< MBL \leq LBUFTP-LBUFLM-1$ [23090]
		MBL + LBUFLM+1	20500B (MBL=50000)	If $MBL \geq LBUFTP-LBUFLM-1$ [23090]
<b>Dump</b>				
7000-6000	1300B	LBUFDP	5300B	
<b>Deadstart</b>				
7000-6000	3300B	LBUFDSPF+LBUFDSLM	16300B	Via permanent file
		LBUFDSTP+LBUFDSLM	16300B	Via tape

Since spun-off tasks are like user jobs, they may be rolled or swapped out as the core is needed.

The relationship between buffer sizes and performance is bound by the same considerations as for any user job. The absolute minimum buffer size is a PRU + 2. Any large reduction in buffer sizes from the release values will have some impact on performance.

† As used with the nominal release definition values.

## ACCOUNTING FOR THE SPOTS

Accounting for the SPOTs is handled through the common deck IPACCT on PL1D. To define accounting on the SPOT job statement, redefine the MICROS ACCTSP (for the spooling SPOT), ACCTTP (for the tape staging SPOT) and ACCTPF (for the permanent file staging SPOT), and leave the MICRO ACCOUNT as a null MICRO string. For accounting on an ACCOUNT statement, redefine the MICROS ACCTSP, ACCTTP, and ACCTPF as before and also redefine the MICRO ACCOUNT to the MICRO string ACCOUNT. For further information and examples, consult the common deck IPACCT. If accounting is not used (that is, IP.ACNT equals 0) then no action is necessary.

The ACCTVAL micro in the common deck IPACCT provides a facility to allow account numbers to be validated and used for files that are cataloged by a SAVEPF from another mainframe. The ACCTVAL micro names an installation-defined program which can be called to validate the AC parameter from the SAVEPF statement that is passed to the SAVEPF SPOT job. The AC parameter is passed to the installation-defined program as a calling parameter. For example, if the ACCTVAL micro is defined to be VALIDAC, the following control statement will be placed in the SAVEPF SPOT job before the COMMENT.ON. statement.

VALIDAC,acparam.

acparam is the AC parameter value in the SAVEPF statement from the linked mainframe.

The installation program should validate the AC parameter and call a helper PP program to place it into W.CPFACT in the job control point area so that it will be associated with the file that is cataloged. If the AC parameter is invalid or absent, the program should issue a dayfile MESSAGE macro with the text ON. to turn on the dayfile transfer, then an appropriate diagnostic, then a dayfile MESSAGE macro with the text OFF. before aborting the SPOT job. The ON. and OFF. messages instruct the station to transfer the intervening messages to the connected mainframe.

The default value for ACCTVAL is COMMENT., so an account validation program will not be called unless ACCTVAL is redefined by the installation.

## SYMMETRIC STATION

### INSTALLATION PARAMETERS

#### A. TABLE SIZE – COMDECK COMTUNE-PL1D

##### NMF

The number of mainframes (6000 or 7000) to which the station may be linked

##### DATAENT

The maximum number of active data streams

##### SFTS

The size of the station file table is equivalent to DATAENT and used by the passive side only.

##### SNTS

The size of the SPOT name table (SNT). The size of the SNT should be the number of DATAENT plus the maximum number of local SPOTs (4) plus the spooling SPOT (1).

##### IDTMAX

The maximum size of the IDT table controls the number of logical IDs a mainframe may have and allows this value minus two logical IDs.

##### MAXSPOTS

A group of parameters that define the default maximum number of active SPOTs of each type that MFSTAT activates at one time

##### SPOOLS

The maximum number of spooling streams

##### DISNAMESIZE

The size of the display name table for the transparent display interface to the 7000. The default value is set to 1, but it must be increased when the display improvements feature is selected.

## SYNTAXSIZE

The size of the syntax extension table for the transparent display interface to the 7000. The default value is set to 1, but it must be increased when the display improvements feature is selected.

## B. POLLING AND RECALL TIMES - COMDECK COMTUNE-PL1D

### MSEC(I)

Used to set recall times. Since recall is given as 1/4 milliseconds, this function multiplies I by 4 so recall becomes I milliseconds.

### SEC(I)

A way of approximating the number of busy station loops in I seconds

### ISEC(I)

Similar to SEC(I) but for the idle station

### TME7000

The delay in seconds of sending the time request to the 7000

### TME6000

The delay in seconds of sending the time request to the 6000

### STA7000

The delay in seconds used by MFSTAT between sending status requests to the 7000

### STA6000

The delay in seconds used by MFSTAT between sending status requests to the 6000

### LLV6000

The delay in seconds used by MFSTAT between sending load leveling requests applies to 6000 only.

### RCL7000

The recall time in milliseconds used by the busy overlay of MFSTAT when communicating with a linked 7000 mainframe

### RCL6000

The recall time in milliseconds used by the busy overlay of MFSTAT when communicating with a linked 6000 mainframe

### TIMEOUT

The length of time in seconds used by MFSTAT before logging out a linked mainframe when communication is lost

### MSGCNT

The length of time in seconds that MFSTAT leaves informative messages on the B display

### LOCPASSTIME

The delay in seconds used by MFSTAT between looking for local GETPF/SAVEPF operations

### BSYLIM

The length of time the busy overlay of MFSTAT delays after sensing an idle condition before going into an idle state

## IDLRCLTM

The recall time used by MFSTAT when the idle overlay is executing

### NOTE

For better response time, lower both the RCL and STA values. To reduce CPU utilization, increase the RCL value.

### CAUTION

If the RCL and STA parameters are too greatly reduced, this may cause STD (the link medium coupler driver) to be locked in.

## LOOP LIM

The delay in seconds used by MFSTAT between checking for a change in busy-to-idle status (controls the frequency with which the busy portion of the station checks its busy status)

## DSDWAIT

The length of time MFSTAT waits for a reply before it rejects a DSD request

## MAXINCOUNT

The frequency with which MFSTAT calls QAC to check the input queues for files to spool when it is idle

## OVLMAX

The maximum time MFSTAT retains the secondary overlay field length after a load of a secondary overlay

## IDLEMAX

The elapsed time in seconds that MFSTAT waits after all spooling activity has completed before swapping out the spooling SPOT

## SPLLIM

The delay in seconds used by MFSTAT after completion of spooling activity before going idle

## IDLETIME

The elapsed time in seconds after which the spooling SPOT attempts to initiate spooling operations

## IDLETIME2

The elapsed time in seconds after which the spooling SPOT initiates new spooling operations when output spooling is taking place

## SPOOLRCL

The recall time in milliseconds used by the spooling SPOT when there is no spooling activity

## C. BUFFER SIZE – COMDECK COMTUNE-PLID

### BUFSIZE

The size of the I/O buffer in the station control point on the passive side

### DAYBUFSIZE

The size of the MFSTAT buffer for processing spot dayfiles on the active side

### RRBUF

The size of the MFSTAT active transmit buffer, the passive receive buffer, the linked staged packet buffer, and the local stage packet buffer

**LRGBUF**

The size of the MFSTAT receive buffer for the active side of the station

**LRGRBUF**

The size of the MFSTAT transmit buffer for the passive side of the station

**LICRBUF**

The length of the MFSTAT buffer used by the INTERCOM queue utility helper

**LBUFLM**

The length of the link buffer used by SPOT jobs for 6000-7000 permanent file staging and 6000-6000 and 6000-7000 I/O spooling

**LBUFSP**

The length of the disk buffer used by the spooling SPOT for 6000-6000 and 6000-7000 spooling of I/O files

**LBUFPF66**

The length of the buffer used to read and write the disk and link files for 6000-6000 permanent file transfers

**LBUFPF76**

The length of the disk buffer used to read and write the disk for permanent file transfers to and from the 7000

**D. LOAD LEVELING – Input (6000 to 6000) DECK SSH-PL1A**

Load leveling provides the capability of distributing the work load among linked 6000 mainframes with common logical IDs. Load leveling is used for jobs in the input queue with destination IDs (ST specified on the job statement) common to both linked mainframes.

A load leveling algorithm (located in deck SSH near SSH.4350) determines whether or not load leveling is performed. The algorithm uses separate parameters (defined near SSH.311) for class 1 (allocatable) and class 2 (nonallocatable) jobs. Load leveling is performed independently for each class, depending upon these parameters.

Class 1 jobs require only immediately allocatable resources, such as memory and disk space. Class 2 jobs require additional resources that cannot be immediately allocated, because resources such as tape drives must be scheduled. For each class, the algorithm determines whether or not all of the required conditions for load leveling are satisfied by the mainframe on which the algorithm is called.

When user direct access (DA) ECS swapping is disabled (IP.ECSW=0), jobs that require DA ECS are assigned to class 2. When IP.ECSW ≠ 0, jobs that require DA ECS but no nonallocatable equipment are assigned to class 7. Maximum and current job counts and load leveling parameters for class 1 also apply to class 7 jobs.

The following parameters are defined.

**ICPFFNT**

The number of free file name tables (FNTs) that must be available for a mainframe to accept a job

**ICPBJTA/ICPBJTN**

The number of additional class 1/class 2 jobs that are allowed to execute must be greater than or equal to this value for a mainframe to accept a class 1/class 2 job.

**ICPRJTA/ICPRJTN**

The number of class 1/class 2 jobs in the input queue that are ready to run must be less than or equal to this value for a mainframe to accept additional class 1/class 2 jobs.

The job control area contains the mainframe status information used by the algorithm to determine whether or not load leveling is to be performed. (Byte C.JCA of pointer P.SCH points to the job control area.)

For class 1 jobs, the load leveling conditions are as follows.

- The number of available FNTs > ICPFFNT.  
(The number of available FNTs is obtained from C.JCEMC.)
- The number of additional class 1 jobs allowed to execute ≥ ICPBJTA  
(The number of additional class 1 jobs allowed to execute is C.JCMXB-C.JCCNB; that is, the maximum number of class 1 jobs allowed to execute minus the number currently in execution equals the number of additional class 1 jobs allowed to execute.)



- The number of class 1 jobs in the input queue that are ready to run  $\leq$  ICPRJTA  
(The number of class 1 jobs in the input queue that are ready to run is obtained from C.JCNJL.)

For class 2 jobs, the load leveling conditions are as follows.

- The number of available FNTs  $>$  ICPFFNT  
(The number of available FNTs is obtained from C.JCEMC.)
- The number of additional class 2 jobs allowed to execute  $\geq$  ICPBJTN  
(The number of additional class 2 jobs allowed to execute is C.JCMTB-C.JCCTB; that is, the maximum number of class 2 jobs allowed to execute minus the number currently in execution equals the number of additional class 2 jobs allowed to execute.)
- The number of class 2 jobs in the input queue that are ready to run  $\leq$  ICPRJTN  
(The number of class 2 jobs in the input queue that are ready to run is obtained from C.JCNTJ.)

When the system invokes load leveling, the sending mainframe transfers jobs with destination IDs to the receiving mainframe for processing. The following conditions must exist on both mainframes for load leveling to be performed.

- Any of the three conditions (for the class being considered) on the sending mainframe is false. (This is normally the case, since the number of free FNTs is usually greater than ICPFFNT.)
- All three conditions on the receiving mainframe are true. (This occurs if enough FNTs are free, enough additional jobs are allowed to go into execution, and only a few jobs are in the input queue; that is, the mainframe can accept more jobs.)

#### E. LOAD LEVELING – Output (6000 to 6000) COMDECK COMTUNE - PL1D

Load leveling for these types of output files takes place only for files with DIDs shared by both machines if there is more than this number for a particular type in the output queue.

ICPTPRT (5)

The load leveling threshold value for print files

ICPTRUN (5)

The load leveling threshold value for punch files

ICPTOTH (5)

The load leveling threshold value for types of files other than print and punch files, such as microfilm, plotter, and so forth

#### F. DSD PARAMETERS – PL1A

IP.LINK

The number of links that can be connected to the station. Used in text IPTEXT, defined at IPARAMS.15

IP.7LNK

For symmetric (6000 to 7000) or reverse (7000 to 6000) staging, IP.7LNK (within DSD at DSD.220) must be defined as not equal to zero; and the symmetric-reverse staging group (SYSTG) (SCOPE 2.1) must be LCM-resident. If SYSTG is disk-resident, then IP.7LNK must equal zero.

IP.STEX

For use of the display improvements interface (which uses the message codes MC.LSYN, MC.LCOM, and MC.LDIS), this parameter must be specified in DSD at IPARAMS.15 as unequal to zero.

#### G. ADDITIONAL PARAMETERS

Refer to COMTUNE and COMTUNX (two common decks on PL1D) for a detailed description of additional station tuning and configuration parameters. The minimum, maximum, and default values are defined together with notes and cautions on each variable.

ECS as a Link Medium Device

For use of ECS as a link medium device instead of the 6683-6683 remote couplers on a 6000 to 6000 link, refer to ECS as a Link Medium Between Mainframes in this section.

## CDC CYBER UTILITIES

The utilities COPYL and ITEMIZE are contained on PL1C. They are described in the CDC CYBER Common Utilities Reference Manual.

The CDC CYBER Utilities installation parameters are shown below with the released default value.

IP.PD	6	Listing lines per inch
LINP	60	Lines per page
LINP8	79D	Lines per page in 8 lines/inch mode

## ACCOUNTING FOR GENLDPF GENERATED JOBS

GENLDPF generates jobs to selectively load permanent files that had a permanent file catalog entry at the time PFLOG was run. Accounting for the generated jobs is handled through the common deck COMSLOG on PL1B. Generated job and account statements are located at Update identifiers GENACNT.1 and GENACNT.2, respectively. Refer to common deck COMSLOG for complete directions for modification of these statements. If accounting is not used (that is, IP.ACNT = 0), no action is necessary.

## GEMINI LOAD LEVELING

Gemini, a control point program, provides input and output file load leveling between two linked mainframes under the control of NOS/BE. An RMS queue set shared between the two mainframes is the only configuration requirement.

Gemini at a control point on one mainframe communicates with Gemini at a control point on the other mainframe through two files cataloged on a shared RMS set (not necessarily the shared queue set). These files contain sections for communication of the following information.

- Contents of the IDT (mainframe and logical identifier table)
- Message from the sending mainframe (used for file transfer)
- Message acknowledgment from the receiving mainframe
- Contents of the input, output, punch, special, and execution queues in QAF format (refer to the NOS/BE System Programmer's Reference Manual)
- Common LID input and output queue counts (common LIDs are those that appear in the IDT for both mainframes)

Only the ownership of the input or output queue files is transferred between the two mainframes because the files exist on a shared queue set. The messages are pointers to the appropriate catalog entry being transferred. The IDT information distinguishes between LIDs unique to one mainframe and those common to the two mainframes. The IDT information is updated whenever Gemini detects a change in the IDT. Queue counts of common LIDs are kept to allow Gemini to determine when load leveling is required. The contents of queues for a mainframe allow INTERCOM users to locate their files through the MYQ/Q/FIND utilities.

## INSTALLATION PARAMETERS

The following parameters can be changed by making Update insertions at GEMIPRM.7. Default values are shown in parentheses following the parameters.

N\$MF CEQU n (1)

Number of mainframes in Gemini network; n can be 0, 1, or 2. A value of 0 or 1 informs the routine MYQ that the Gemini link is not available. N\$MF must be equated to 2 if Gemini is to be used.

PID\$MFA CMICRO 1,,/xxx/ (MFA)

PID\$MFB CMICRO 1,,/yyy/ (MFB)

xxx and yyy are physical identifiers (PIDs) of the two mainframes in the network. They must be three letters and must match the PID in the IDT.

PID\$PFNA CMICRO 1,,/xx.../ (LINKFILE)

PID\$PFNB CMICRO 1,,/yy.../ (LINKFILE)

Permanent file names of the two link files in the network (one through forty characters).

PID\$IDA CMICRO 1,,/xxx.../ (MFASYSTEM)  
PID\$IDB CMICRO 1,,/yyy.../ (MFBSYSTEM)

Permanent file identifiers (one through nine characters) of the two link files in the network. If these parameters are not specified but the PID\$MFx micros are specified, the defaults are the characters SYSTEM prefixed with the PID\$MFx parameters.

LINK\$AC CMICRO 1,,/xxx.../ (123456789)

Account number (one through nine characters) of the two link files in the network.

LINK\$SN CMICRO 1,,/xxx.../ (\*PF)

Set on which the link files reside. Possible values are \*PF, \*Q, \*SYS, or the actual setname. The link files must reside on a shared RMS set.

LL.RCL CEQU n (6)

Gemini recall count; delays Gemini execution. When Gemini finds unique LIDs with files to transfer, it loops every half second to complete a portion of the transfer process. When nothing remains to be done and there are no more files to transfer, Gemini waits LL.RCL\*.5 seconds before looping, thus saving system resources. The queue information for one queue on the link file is updated every LL.RCL\*.5 seconds; therefore, it takes 5\*LL.RCL\*.5 seconds to update all queue information on the link file. Gemini attempts to initiate load leveling for common LIDs every 5\*LL.RCL\*.5 seconds. LL.RCL can be used to adjust the balance between Gemini processing and system utilization. To increase Gemini processing, reduce the value; to reduce system utilization, increase the value.

LL.MINI CEQU n (2)  
LL.MINO CEQU n (2)

Minimum count for load leveling input (I) and output (O) files (refer to Load Leveling Algorithm).

LL.MAXI CEQU n (6)  
LL.MAXO CEQU n (6)

Maximum count for load leveling input (I) and output (O) files (refer to Load Leveling Algorithm). LL.MAXx should be at least three times LL.MINx to avoid transferring too many files.

#### LOAD LEVELING ALGORITHM

Gemini makes load leveling decisions based upon comparisons of input file counts or output file counts for one common LID at a time. (Output file count includes all files with a disposition code of 10 or greater.)

Gemini transfers LL.MINx files for a common LID to the second mainframe if the first mainframe has at least LL.MAXx files and the second mainframe has fewer than LL.MINx files. Gemini also transfers one file for a common LID to the second mainframe if the first mainframe has at least two files but fewer than LL.MAXx files and the second mainframe has no files.

In addition, Gemini transfers files with an LID that exists on only one mainframe. It scans tables for this type of transfer every LL.RCL\*.5 seconds whenever it is not busy load leveling files.

#### NOTES AND CAUTIONS

Gemini does not discriminate among job classes when the input files have the same LID. Gemini does not check for dependency IDs when load leveling is executing. It is suggested that the installation set up LIDs based on job statement parameters such as the following:

- Time use
- Memory use
- ECS use
- Tape use
- RMS use
- I/O use
- Priority level
- Dependency

The use of dependency requires a unique LID. The use of tapes might also require a unique LID. The other parameters could be grouped and given a unique LID or a common LID, depending upon the installation goals.

Gemini and the station can be run on the same system at the same time. However, the station predominates over Gemini in transferring jobs.

#### DEADSTART DUMP ANALYZER

PL1B, PL12, and PL14 contain the fast dump analyzer. The overlays on PL12 and PL14 are for INTERCOM analysis only. If neither INTERCOM 4 nor INTERCOM 5 is to be built, an analyzer without the capability to format INTERCOM tables can be produced by inserting the following directive after the line that states ADD MODIFICATIONS HERE in the second input record of PL1B:

##### **\*DEFINE NOINT**

This modification removes IT as a default parameter value and the resulting analyzer flags the IT analyzer option as an error. All other options are unaffected.

The directions for taking an express deadstart dump (EDD) appear in the NOS/BE Operator's Guide, and the directions for using the analyzer appear in the NOS/BE System Programmer's Reference Manual.

### RELEASE DESCRIPTION

COMPASS is a comprehensive assembler program for the CDC CYBER 170, CDC CYBER 70, 6000, and 7000 Series computer systems. COMPASS 3 runs under NOS/BE 1 and requires the same minimum hardware configuration as NOS/BE 1.

The common common decks are a set of debugged COMPASS subroutines that perform such functions as:

- Data conversion
- Dynamic table management
- Register saving and restoration
- I/O interface with CIO and FET

### RELEASE MATERIALS

The release tape for COMPASS, PL2, contains a source program library as file 1 and assembled binary as file 2.

### GENERAL DESCRIPTION

COMPASS consists of three overlays. The level (0,0) overlay COMPASS is the main control program. The level (1,0) overlay COMP3\$ contains the assembler which can be called by compilers to process embedded COMPASS source programs. The level (1,1) overlay COMP3\$A contains the part of the assembler that is loaded after initialization is complete.

The common common decks are available as UPDATE COMDECKS in source form on the COMPASS oldpl or as relocatable subroutines in the SYSLIB library.

### INSTALLATION PARAMETER

To ensure efficient code generation, the MODEL micro in deck IPARAMS on PL1A must be set to the proper value for the target machine.

COMPASS has one installation parameter, CP.OVLIB, the library name for overlays. In the released system, overlays COMP3\$ and COMP3\$A must be in a library in the global library set for a job or in the NUCLEUS library. COMPASS loads its overlays from a specific library with the following.

```
*D CPS028.10
CP.OVLIB      MICRO 1,*,libname*
```

Changing this parameter from its default state necessitates change in the installation deck EDITLIB record. Note that this change is needed only when the overlays are to be in a system library other than NUCLEUS. If the micro CP.OVLIB is left null (as released), COMPASS can be executed without change or reassembly from the system library NUCLEUS, from any system or user library named on a LIBRARY control statement, or from an overlay (nonlibrary) file.

### INSTALLATION PROCEDURES

Installation of COMPASS and the common common decks requires obtaining job decks PL2I and PL2E from the installation deck program library tape as outlined in part I, section 1 of this document.

PL2I is a maintenance deck used to create a revised program library and binary file. PL2E can be used to enter COMPASS and the common common decks into the running system or user libraries from either the released PL2 or a tape created by PL2I. After deck PL2E has completed, job DST3 can be run to create a deadstart tape of the running system. Job decks PL2E and DST3 need not be run if the user library installation process is being followed.



# CDC CYBER RECORD MANAGER I BASIC ACCESS METHODS

## RELEASE DESCRIPTION

CDC CYBER Record Manager Basic Access Methods (BAM) consists of modules for creating, updating, and accessing two file organizations: sequential (SQ) and word-addressable (WA).

BAM 1.5 operates under NOS/BE on the same minimum configuration as NOS/BE.

The structure of the release tape PL3A is as follows.

<u>Files</u>	<u>Content</u>
1	Program library in UPDATE format
2	TXTCRM, IOTEXT, SYSTEXT binary
3	Control modules binary
4	Encapsulated modules binary
5	FILE, CRMEP control statement processor relocatable binary
6	FILE, CRMEP control statement processor absolute binary
7	FORTTRAN Extended 4 and FORTTRAN 5 interface binary

## NOTES AND CAUTIONS

The display option on parity errors is not implemented.

If C-blocked, non-W record, SI tapes are copied to S tapes, section boundaries may be lost.

End of block padding is not supplied on last block (short block) of each partition.

## ADDITIONAL INFORMATION

SQ/WA I/O modules are divided into two parts: Initialization Modules and Sequential and Word Addressable I/O Modules.

### INITIALIZATION MODULES

These routines control selective loading based on file organization. They contain jump vectors directing a user call to the I/O appropriate to the file organization selected. Their program names have an RM suffix.

### SEQUENTIAL AND WORD ADDRESSABLE I/O MODULES

The I/O macro text included with the SQ/WA program library is IOTEXT, which is identical to the default SYSTEXT. It consists of, but is not limited to the macros included in the following table. (Some auxiliary macros exist which are not supported at the user level.)

<u>Macro Name</u>	<u>System</u>	<u>Reference</u>	<u>Comdeck</u>
FILE	<u>CRM</u>	<u>CDC CYBER Record Manager Reference Manual</u>	<u>CRMCOM</u>
FETCH	↓	↓	↓
FLUSHM			
STORE			
OPENM			
CLOSEM			
GET			
GETP			
GETN			
GETNR			
GETWR			
PUT			
PUTP			
PUTWR			
REPLACE			
DELETE			
ENDFILE			
SKIPdu			
d=F/B,u=L/P/F			
SEEK			
START			
REWINDM			
WEOR			
WTMK			
GETL			
PUTL			
CLOSEL			
CHECK	<u>NOS/BE 1</u>	<u>NOS/BE 1 Reference Manual†</u>	<u>ACTCOM</u>
ABORT	↓	↓	↓
CHECKPT			
CLOCK			
CONTRLC			
DATE			
DISPOSE			
ENDRUN			
FILESTAT			
IO TIME			
IXi Xj/Xk			
IXi Xj/Xk,Bn			
JDATE			
LOADREQ			
MEMORY			
MESSAGE			
RECALL			
RECOVR			
REQUEST			
ROUTE			
RTIME			
STATUS			
SYSCOM			
SYSTEM			
TIME			
TRANSR			

† These macros are source compatible with the corresponding macros on CPCTEXT but they do not generate the same code.



## INSTALLATION PARAMETERS

To ensure correct code generation, the MODEL micro in deck IPARAMS on PL1A must be set to the correct value for the target machine.

The following installation parameters permit a certain amount of tailoring.

<u>Parameter</u>	<u>Mnemonic Update ID</u>	<u>Description</u>	<u>Default Value</u>
*DELETE LBLIM.1	LBLIM	Length of label buffer and size limit of user label string. Each user label requires 9 words. LBLIM should be n*9+1, where n is the maximum number of labels permitted (HDR1-9, . . .).	10 decimal

Use of the compare/move unit hardware is affected by the MODEL micro definition. If HF.C is defined, SQ/WA routine MOVE\$RM assembles using the CMU hardware. For records over 40 characters, the CMU hardware reduces CP time of a program using SQ/WA.

The MODEL micro definition can be overridden in an update of PL3 as follows:

To run CMU on:

```
*D      F2950CR.17,21
≡CMU≡   EQU 1
*C      TXTCRM
```

To turn CMU off:

```
*D      F2950CR.17,21
≡CMU≡   EQU 0
*C      TXTCRM
```

## INSTALLATION PROCEDURES

File 1 of PL3A contains the SQ/WA program library.

Files 2 through 7 are preassembled binaries assembled with default installation parameters.

Installation of SQ/WA requires obtaining job deck PL3AI, PL3AE, and PL3AO from the installation deck program library tape, as outlined in part I, section 1.

Deck PL3AI references IPTEXT. Part III of this document contains a cross reference map of IPARAMS symbols and routines that reference these symbols.

Deck PL3AI is a maintenance deck which allows regeneration of PL3A. This deck updates the program library, assembles SQ/WA, and places the binary on the new PL as supplemental files. User-selected installation parameters should be modified at the indicated place in PL3AI. Deck PL3AI requires access to the NOS/BE 1 program library PL1A to acquire the common decks ACTCOM and COMSRAS used by the CRM system texts.

Deck PL3AE adds SQ/WA to the running system or user libraries, either from the released PL3A or a PL3A created by deck PL3AI. Then deck DST3, described in section 1, can be run to create a deadstart tape of the running system.



RELEASE DESCRIPTION

CDC CYBER Record Manager Advanced Access Methods (AAM) includes modules for creating, updating, and accessing Indexed Sequential (IS), Direct Access (DA), Actual Key (AK), and Multiple Index Processor (MIP) files. CRM AAM exists in two versions, Initial and Extended. AAM Extended supports only Extended IS and Extended MIP.

AAM INITIAL

A key analysis utility routine (KYAN) is available to aid in the selection of a hashing routine for direct access files.

A create utility (CREATE) is available for efficiently creating DA files.

Both KYAN and CREATE utilities require CMM to be available.

Two utility routines can be called by control statements for indexed sequential files: SISTAT prints the statistics for an existing IS file; ESTMATE produces estimates of block and buffer sizes from input statements containing IS file descriptions.

IXGEN, a utility routine to invert any existing AAM Initial file, provides an efficient method of creating multiple access paths for a file.

With AAM encapsulated, it is no longer possible to separate portions of the Initial IS file processor (for example, across overlays). The entire Initial IS processor is one capsule, and the individual IS routines are no longer distinguishable. Operating under dynamic loading mode (with FDL), the capsules can be selectively loaded as needed, so it is not necessary to build any capsules into specific overlays. CTL\$RM and CTRL\$AA, through which control flows from the user program to AAM, should be included in the 0,0 overlay. It is possible to switch overlays without closing all AAM files first; however, because capsules are loaded above highest high address (HHA), field length will remain high if files are left opened, since the capsules will not be unloaded. If the capsules are to be built statically into an overlay, the whole capsule must be in the same overlay.

AAM EXTENDED

Four utility routines called by control statements are provided for AAM Extended files. FLSTAT prints the statistics for existing files; FLBLOK produces estimates of block and buffer sizes from input statements; MIPGEN inverts an existing data file on any number of alternate keys and produces or modifies the associated MIP file; MIPDIS disassociates or reassociates MIP and data files.

The utility routines KYAN and CREATE (described previously) are also available for AAM Extended files.

RELEASE MATERIALS

The direct access, actual key, initial indexed sequential, and initial multiple index processor modules are contained on release tape PL3B. Extended IS and Extended MIP are contained on release tape PL3C.

The structure of the release format PL3B tape is as follows.

<u>File</u>	<u>Content</u>
1	Program library in UPDATE format
2	Encapsulated I/O modules
3	Absolute binaries for the AAM Initial utilities
4	Binaries for the AAM Initial relocatable programs
5	Binary for MSD PP routine
6	Relocatable binaries for IXGEN
7	Relocatable binaries for ESTMATE and SISTAT

The structure of the release format PL3C tape is as follows.

<u>File</u>	<u>Content</u>
1	Program library in UPDATE format
2	Encapsulated I/O modules
3	Absolute binaries for the AAM Extended utilities
4	Binaries for the AAM Extended relocatable programs
5	Relocatable binaries for MIPGEN
6	Relocatable binaries for FLSTAT
7	Relocatable binaries for FLBLOK
8	Relocatable binaries for MIPDIS

### LIMITATIONS

The CREATE, IXGEN, and MIPGEN utilities require that Sort/Merge be installed. If Sort/Merge is not available, comparable DA and multiple-indexed files can be created through explicit CRM calls.

### AAM INITIAL INSTALLATION PARAMETERS

The system contains parameter values that are effective when the user does not supply settings. The default parameters are defined on the program library tape PL3B. For jobs used to change these parameter settings, refer to part I, section 1.

IS PARAMETERS (located in common deck SISCOMM)

<u>Parameter</u>	<u>Released Default Value</u>	<u>Significance</u>
DFBKGFAC EQU	2	Default data record blocking factor Range: 1 through 4095 Location: SISCOMM.17
DFDAPADP EQU	0	Default data block padding factor Range: 0 through 99 Location: SISCOMM.18
DFIBKSZW EQU	511	Default index block size in words Range: 1 through 23,767 Location: SISCOMM.20
DFINPADP EQU	5	Default index padding factor Range: 0 through 99 Location: SISCOMM.21
DFNRLVLS EQU	1	Default number of index levels Range: 1 through 63 Location: SISCOMM.22
KEYLIMIT EQU	255	Maximum key size in characters Range: 1 through 511 Location: SISCOMM.32
TOTFILES EQU	10	Maximum number of active IS files per run (specifies an internal table size in words). System file limit. Location: SISCOMM.277

DA PARAMETERS (located in common deck SDACOM)

<u>Parameter</u>	<u>Released Default Value</u>	<u>Significance</u>
DBLK EQU	2	Default blocking factor. Limited by user field length. Location: SDACOM.40
NMOPN EQU	10	Maximum number of DA files allowed to be opened concurrently. System file limit. Location: SDACOM.41

AK PARAMETERS (located in common deck SAKCOM)

<u>Parameter</u>	<u>Released Default Value</u>	<u>Significance</u>
PRCNTBL EQU	25	Percent required for random probing Range: 0 through 100 Location: SAKCOM.37
PROBLEM EQU	4	Number of random probes. No limit. Location: SAKCOM.38
DBLK EQU	8	Default blocking factor. Limited by user field length. Location: SAKCOM.39
NMOPN EQU	10	Maximum number of opened AK files. System file limit. Location: SAKCOM.42

IS READ-ONLY PARAMETERS (located in common deck ROCOM)

<u>Parameter</u>	<u>Released Default Value</u>	<u>Significance</u>
TOTFILES EQU	10	Maximum number of files allowed to be opened concurrently for processing by IS read-only processor. System file limit. Location: ROCOM.119

DA READ-ONLY PARAMETERS (located in common deck RODCOM)

<u>Parameter</u>	<u>Released Default Value</u>	<u>Significance</u>
OFCOUNT DATA	10	Maximum number of files opened concurrently for processing by DA read-only processor. No limit. Location: RODCOM.31

AAM EXTENDED INSTALLATION PARAMETERS

When installing AAM Extended, code to gather additional file statistics is assembled if the UPDATE directive \*DEFINE STATS is included in the UPDATE input to the AAM Extended program library. If this directive is omitted, only normal file statistics are gathered.

USER ADDITIONS TO AAM EXTENDED

AAM Extended includes one system compression/decompression routine. A site can add up to 53 additional user written compression/decompression or encoding routines as system routines. Each added routine must be encapsulated and the capsule OPEN\$AA must be modified. The procedure to add routines follows.

The routine must have one entry point whose name is of the form CMPR\$nn, where nn is two decimal digits in the range 11 through 63. The entry point name of the first routine added must be CMPR\$11, the second must be CMPR\$12, and so on. The entry point must be the second word (word 1) of the routine.

The first three words of each routine must have the following format.

<u>Word</u>	<u>Bits</u>	<u>Contents</u>
0	59-18 17-0	Entry point name, display code, left-justified with zero fill 1
1	59-18 17-0	0 Starting address of compression code
2	59-18 17-0	0 Starting address of decompression code

The following illustrates the construction of a single site-added compression/decompression routine.

```

IDENT
ENTRY  CMPR$11
VFD    42/0LCMR$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 CDC CYBER Loader requires standard relocation for fast dynamic loading of capsules; therefore, the VFD statements must be constructed as shown in the preceding example. Execution of the compression or decompression code is effected by a return jump to the address specified in word 1 or word 2 of the routine.

An entry must be added to the capsule name table in deck OPNMDAA for each added routine. The macro GENTBL (also part of OPNMDAA) generates the table entry and has the following calling format.

```

GENTBL epname
      epname  Entry point name specified in word 0 of added routine

```

Table entries must be specified in consecutive, ascending numerical order. For example, if three routines are added, the following change to OPNMDAA must be made.

```

*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 sequence of control statements including the following.

```

.
.
.
UPDATE (K)
.
.
.
COMPASS(I,S=TXTCRM,S=IPTEXT)           Assembles OPNMDAA and DICODAA
SYMPL(I,LXR)                           Compiles OPENDAA
COMPASS.                                Assembles user-written routine
.
.
.
GROUP($AAM$SCTL$)                       Encapsulates the modified OPNM$AA capsule and the new com-
CAPSULE($OPNM$AA$)                       pression capsule
CAPSULE($CMPR$$I$)
LDSET(OMIT=$SETUP,$/$RM$$SYS=$)
LOAD,LGO.
NOGO,NEWCAP.
EDITLIB,SYSTEM.
7/8/9
*IDENT
.
.
.
    UPDATE directives to modify OPNMDAA
.
.
.
*C OPNMDAA,DICODAA,CWEORI,OPENDAA
7/8/9
.
.
.
    User routine being added
.
.
.
7/8/9
READY(SYSTEM)
LIBRARY(AAMLIB,OLD)
REPLACE(*,NEWCAP)
FINISH.
COMPLETE.
ENDRUN.
7/8/9
6/7/8/9

```

#### INSTALLATION PROCEDURES FOR AAM INITIAL

To ensure proper code generation, the MODEL micro in deck IPARAMS on PL1A must be set to the proper value for the target machine.

PL3B contains seven files. File 1 contains the program library which will generate 10 COMPILE file records.

<u>Source Language</u>	<u>Description</u>
COMPASS	Code for PP diagnostic routine
COMPASS	Code for KYAN and CREATE utilities (relocatables to NUCLEUS library)
COMPASS	Code for IS, DA, AK, and multiple index CP routines
SYMPL	Code for multiple index CP routines
COMPASS	Code for IXGEN utility
SYMPL	Code for IXGEN utility
FORTTRAN Extended 4	Code for SISTAT and ESTMATE utilities (overlays to NUCLEUS library)
COMPASS	Code for first skeleton used by COPYL to rearrange binaries for capsule formation
COMPASS	Code for second skeleton
COMPASS	Code for third skeleton

File 2 contains capsules for the AAMLIB library; files 3 and 4 contain binaries for the NUCLEUS library; file 5 contains binaries for the PP library.

PL3BI is a maintenance deck which can be used to create a revised program library and binary file containing modifications. PL3BE can be used to enter AAM Initial into the running system or user libraries, either from the released tape or a tape created by PL3BI. Job DST3 can be run to capture a deadstart tape containing AAM Initial. Decks PL3BE and DST3 need not be run if the user library installation process is being used. In non-ULIB mode, PL3BI executes an EDITLIB to the running system AAMLIB library to make required routines available during further overlay generation. PL3BO allows regeneration and replacement of the absolute overlays on the tape PL3B, using the tape written by PL3BI.

Successful assembly of DA requires that Sort/Merge be previously installed (refer to Limitations). If SMTEXT is not present in the running system when deck PL3BI is run, three assembly errors will appear in routine DCREATE, and deck PL3BE will yield an EDITLIB diagnostic.

**NOTE**

If FORM calls the KYAN utility from an owncode exit, KYAN must be available in the system.

INSTALLATION PROCEDURES FOR AAM EXTENDED

PL3C contains eight files. File 1 contains the program library which will generate COMPILE file records.

<u>Source Language</u>	<u>Description</u>
SYMPL	Code for Extended IS, DA, AK, and Extended MIP CP routines
COMPASS	Code for Extended IS, DA, AK, and Extended MIP CP routines
COMPASS	Code for MIPGEN utility
COMPASS	Code for skeleton file used by COPYL to rearrange binaries for capsule formation
FORTRAN Extended 4	Code for FLBLOK utility
SYMPL	Code for FLSTAT utility
COMPASS	Code for FLSTAT utility
SYMPL	Code for MIPDIS utility
COMPASS	Code for MIPDIS utility
COMPASS	Code for static load subroutines

PL3CI is a maintenance deck which can be used to create a revised program library and binary file containing modifications. PL3CE can be used to enter AAM Extended into the running system or user libraries, either from the released tape or a tape created by PL3CI. Job DST3 can be run to capture a deadstart tape containing AAM Extended. Decks PL3CE and DST3 need not be run if the user library installation process is being used. In non-ULIB mode, PL3CI executes an EDITLIB to the running system AAMLIB library to make required routines available during further overlay generation. Deck PL3CO allows regeneration and replacement of the absolute overlay file on tape PL3C, using the tape written by PL3CI.

VERIFICATION PROGRAMS

FORTRAN Extended 4 and COBOL must be installed before the corresponding installation verification program can be run. Comment statements describe the purpose of each deck. Eight verification programs are provided.



RELEASE DESCRIPTION

8-Bit Subroutines run under NOS/BE and the CDC CYBER Record Manager.

RELEASE MATERIALS

The 8-Bit Subroutines are released on release tape PL4 together with FORM. A complete catalog of PL4 contents follows.

Files 1 through 3	8-Bit Subroutines
1	Program library in UPDATE format
2	8-Bit Subroutines binary capsules
3	COPY8P absolute binary
Files 4 through 7	FORM
4	Program library in UPDATE format
5	Relocatable capsules (run time system)
6	FORM main overlay relocatable binary
7	FORM main overlay absolute binary

HARDWARE CONFIGURATION

The 8-Bit Subroutines require the same minimum hardware configuration as NOS/BE. An extended print train is required to print ASCII 96-character graphic files, if used.

GENERAL DESCRIPTION

The relocatable routines from the 8-Bit Subroutines run under NOS/BE and CDC CYBER Record Manager with COBOL, FORTRAN Extended, or COMPASS. COPY8P, a stand-alone routine used to print 360/370 files, can be called from a COPY8P control statement and runs under NOS/BE.

INSTALLATION PROCEDURES

Part III contains a cross mapping of referencing routines and IPARAMS symbols found in IPTEXT.

Installation job decks PL4AI, PL4AE, PL4AV1, and PL4AV2 can be obtained from the Installation Deck program library using the procedure described in part I, section 1.

Deck PL4AI is a maintenance deck that allows updates of the 8-Bit Subroutines on the PL4 tape. This deck updates the program library, assembles the relocatable object routines, assembles COPY8P, and creates a new COPY8P absolute overlay. The job allows creation of a revised PL4 release tape.

Deck PL4AE adds the 8-Bit Subroutines to the running system. Relocatable object routines are put in the BIT8LIB library. COPY8P becomes part of the NUCLEUS library. Deck DST3 then can be run to create a deadstart tape of the running system. Decks PL4AE and DST3 need not be run if the user library installation process is being followed.



---

**GENERAL DESCRIPTION**

FORM copies and restructures data files. It can be called from a FORM control statement.

**RELEASE MATERIALS**

FORM is contained on release tape PL4 with the 8-Bit Subroutines. A complete catalog of the PL4 contents appears in part II, section 5.

**HARDWARE CONFIGURATION**

FORM requires the same minimum hardware configuration as NOS/BE.

**INSTALLATION PARAMETERS**

No IPARAMS are used.

PL4 contains seven files; files 4 through 7 pertain to FORM.

Installation job decks PL4BI, PL4BE, and PL4BO may be obtained from the Installation Deck program library using the procedure described in part I, section 1.

Deck PL4BI is a maintenance deck that allows updates of FORM on the PL4 tape. This deck updates the library, assembles and encapsulates the relocatable routines, and assembles and builds a new FORM main overlay. The job allows creation of a revised PL4 release tape.

PL4BE adds FORM to the running system. Relocatable capsules are placed in the BIT8LIB library. FORM is placed in the NUCLEUS library.

Deck PL4BO allows regeneration and replacement of the absolute overlay file on tape PL4.



## RELEASE DESCRIPTION

The On-Line Maintenance Software (OLMS), previously known as the CE Diagnostics, requires the same hardware configuration as NOS/BE.

## RELEASE MATERIALS

The On-Line Maintenance Software is released on the release tape PL5A.

The structure of the program library tape is as follows:

<u>File</u>	<u>Content</u>
1	Program library in Update format
2	PP COMPASS binaries
3	CP COMPASS binaries
4	FORTRAN binaries

## INSTALLING DIAGNOSTIC PROGRAMS

Installation of this product requires the Update directive \*DEFINE, names to write correct data to the COMPILE file and the F parameter on the COMPASS statement to control code assembly for proper binary creation. To include test CMS for a mainframe with a compare/move unit, place a \*DEFINE, CMU directive in job PL5I. To install MALET drivers, a DEFINE must be made for each required driver as follows:

<u>DEFINE Directive</u>	<u>Required Driver</u>
*DEFINE,580	580 driver for 512/580
*DEFINE,MUX	Multiplexer driver 6671/6676/2550-100
*DEFINE,65X	60x/65x driver
*DEFINE,66X	66x driver
*DEFINE,67X	67x driver
*DEFINE,841	841 driver
*DEFINE,844HT	844 half track driver
*DEFINE,844FT	844 full track driver
*DEFINE,SECURE	Read preallocated area only unless disk is logically idle or unloaded
*DEFINE,FMDHT	885 half track driver
*DEFINE,FMDFT	885 full track driver

Both drivers are required for 7x5x controllers

Both drivers are required for 885 disk drives

The COMPASS F parameter must be equated to 4 (F=4) when generating binaries of COMPASS programs.

## INSTALLATION PARAMETERS

Release values of the installation options in the deadstart diagnostic sequencer are as follows.

NOISEL (DDS.194)

Same as IP.NOISE in deck CIOCOM on PL1A; default setting is 3.

Refer to the On-Line Maintenance Software Reference Manual for a description of the options available.

## INSTALLATION PROCEDURES

Installation of the On-Line Maintenance Software requires deck PL5AI from the installation deck program library as outlined in part I, section 1.

Deck PL5AI references IPTEXT; part III of this document contains a cross-reference map of IPARAMS symbols and routines that reference these symbols.

Installation job PL5AI updates and creates a new program library including assembled/compiled binaries.



## RELEASE DESCRIPTION

Maintenance tools for NOS/BE are provided on release tape PL6. These maintenance tools are divided into two categories: SYMPL compiler and conversion aids. The structure of the release format PL6 is as follows.

Files 1 through 4	SYMPL
1	SYMPL source in UPDATE program library format
2	SYMPL compiler in relocatable binary
3	SYMPL compiler in absolute overlay binary
4	SYMPL object library in relocatable binary
Files 5 through 6	Conversion aids
5	Conversion aids source in UPDATE program library format
6	Conversion aids binary

## SYMPL

SYMPL (Systems Programming Language) is designed to facilitate systems programming; it does not contain some features normally found in higher level languages, such as complex arithmetic and input/output capability. Instead, it contains features particularly suited to systems programming, such as bit manipulations, based arrays, and an elementary macro capability. It produces code optimized for efficient register and functional unit usage, particularly oriented toward the 6600-type mainframe.

The SYMPL compiler is written mainly in SYMPL; only the system interface routines are in COMPASS. Thus, an absolute binary of SYMPL is necessary for installation if changes are to be made to the source.

## INSTALLATION PARAMETERS

SYMPL has no installation parameters. However, to ensure efficient code generation, the MODEL micro in deck IPARAMS on PL1A must be set to the proper value for the target machine.

## INSTALLATION PROCEDURES

Before SYMPL can be installed, NOS/BE, COMPASS, and the FORTRAN Extended 4 object library must have been installed. SYMPL can be updated and installed with the following jobs. Job PL6AI updates the SYMPL library tape. Job PL6AE edits SYMPL into the system from a SYMPL program library tape. Deck PL6AO allows regeneration of absolute overlays plus creation of a new PL6 tape. Because SYMPL is written in its own language (SYMPL), it is recommended that the latest available SYMPL binaries be present in the running system before installing SYMPL. The unconfigured deadstart tape base system will always contain the latest SYMPL binaries; all other base systems should enter the latest available SYMPL binaries into the running system using deck PL6AE.

## VERIFICATION PROGRAM

The best verification of successful installation of SYMPL is satisfactory compilation of FORM and QUERY UPDATE.

## CONVERSION AIDS

The following conversion aid programs are provided on the maintenance tools tape.

SIFT	A program to convert FORTRAN 2.3 programs to FORTRAN Extended 4 format
SPY	Utility package used to monitor the P register of a CPU program and provide a histogram of elapsed time used in specific areas of code
DOCK	Utility for extracting IMS information from the NOS/BE program library
STIMULATOR	Utility package used to simulate live INTERCOM 4 low-speed terminals

SIS63	A program to convert a SIS file created on a SCOPE 3.3, 63-character set system, so that it can be processed on a NOS/BE, 64-character set system
CIA	A PPU program used to collect data on CPU utilization or CPMTR execution for performance analysis
CPMET	A program to collect and report detailed data on CPMTR execution

## INSTALLATION PROCEDURES

Job PL6BI updates and compiles file 5 creating a complete revised PL6. Job PL6BE adds binary from file 6 to the running system. These jobs may be obtained from the installation deck program library tape, using the procedure outlined in part I, section 1.

Once PL6BE has completed, job DST3 may be run to create a deadstart tape of the running system. PL6BE and DST3 are not required if the user library installation process is being used.

## USAGE OF CONVERSION AIDS

Usage instructions for SIFT can be found in SIFT (FORTRAN Translator Program) PSB.

Usage instructions for SPY, CIA, and CPMET can be found in the SCOPE V3.3 to SCOPE V3.4.x Conversion Aids PSB.

Usage instructions for the other conversion aids are as follows.

## DOCK USAGE

DOCK is a FORTRAN Extended 4 source language utility for extracting listable internal maintenance specification information from a COMPILE file generated from the NOS/BE program library.

The control card directive is in the following form.

DOCK(p1,p2,p3, . . . ,pn)

### Definition:

Default, if parameter is not specified

Assumed, if parameter is specified, but not equivalenced

I	Name of program source file (assumed to be an update COMPILE or source file not exceeding 90-column BCD characters). Default = SOURCE, assumed = COMPILE.
L	Name of file containing documentation list (cannot be the same name as I). Default = assumed = OUTPUT.
F	Up to 25 characters to be printed in the bottom left corner of each page of documentation Default = INT, Folio = \$INTERNAL DOCUMENTATION.\$ EXT, Folio = \$EXTERNAL DOCUMENTATION.\$ Assumed = \$I M S.\$
INT	Internal; all internal, external, and overview documentation will be listed on file L.
EXT	External; external and overview documentation will be listed on file L. Default = INT.
OVR	Overview; only overview documentation will be listed on file L.
INDEX	At the end of each routine processed an index is printed, all symbols found in location field of EJECT, SPACE, TITLE, and TTL cards. Default = INDEX off.
NR	No rewind of input file (I parameter); default = rewind of INPUT.
NT	No table generation. Default = table generation.
NP	No propagation of page numbers across routine. Default = on.
TE	Documentation file, L, formatted for input into program TEXTJAB. Default = off.



Default parameter settings include the following.

DOCK(I=SOURCE,L=OUTPUT,F=\$INTERNAL DOCUMENTATION.\$,INT,NP)

Assumed parameter settings include the following.

DOCK(I=COMPILE,L=OUTPUT,F=\$INTERNAL DOCUMENTATION.\$,INT,NP)

The following dayfile messages are issued by DOCK.

FL TOO SHORT FOR DOCK. (REQUIRES 12K).

Not enough field length was allowed; current minimum field length is 12K (octal).

FILE NAME CONFLICT.

Input, I, and List, L, file names are the same.

MEMORY OVERFLOW IN BUILDING INDEX TABLE.

Not enough field length for index table; increase by 4K (octal).

EMPTY INPUT FILE. NO DOCUMENTATION PRODUCED.

Input file was empty.

INPUT FILE NAME IS ILLEGAL.  
OUTPUT FILE NAME IS ILLEGAL.

Illegal character specified in file name.

FILE EQUIVALENCE MAY NOT BE 0.

A file parameter cannot be set to zero.

## STIMULATOR OPERATING INSTRUCTIONS

### Automatic Table Setting

The STIMULATOR automatically sets the EST and mux-subtables if the user so specifies. The new operating procedures are as follows.

Read in the SIP job to initiate the STIMULATOR.

After requesting appropriate tape assignments, the STIMULATOR asks

DO YOU WANT AUTOMATIC TABLE SETTING — N.YES OR N.NO

When the answer is YES, the STIMULATOR makes the following checks.

The STIMULATOR checks for a mux-subtable pointer in low core. If no pointer is found, the following diagnostic is issued and the SIP job drops.

ERROR—DONT HAVE MUX SUBTABLE DEFINED

If this check is passed the STIMULATOR searches the EST until it finds a YC entry (turned OFF). When not enough YC entries are found in the EST, the following is displayed.

NOT ENOUGH EXISTING EST YC ENTRIES—SET MANUALLY

EST YC entries will not be created because a valid mux-subtable pointer cannot be chosen arbitrarily.

When a valid YC entry is found, the channel number from the SIP statement C parameter is used.

The mux-subtable pointer from this EST entry is used to modify the appropriate mux-subtable. The SIP card T parameter is placed in the mux-subtable to indicate the number of interactive terminals.

The mux EXT entry is set to indicate a STIMULATOR run. When INTERCOM finds a mux with the STIMULATOR flag set in the EST, it uses a special driver, 1Z8.

When all tables are set properly, the following message is displayed.

—OFF UNDESIRE MUXS - BRING UP INTERCOM.

When INTERCOM is brought up, the following is displayed, and the STIMULATOR run is continued.

CONTINUE SIMULATION

The STIMULATOR changes only the first one or two EST entries encountered (depending on M parameter of SIP statement).

If NO is the answer to DO YOU WANT AUTOMATIC TABLE SETTINGS, the following is displayed.

YOU ARE RESPONSIBLE FOR TABLES SETUP—TYPE N.GO TO CONTINUE

Thus, the user can set tables manually if desired.

#### Manual Table Setting

The EST must contain an entry for a 6671 or 6676 multiplexer specifying the channel to be used for the stimulation. The entry should point to a valid multiplexer subtable. The entry can be typed manually after unlocking the keyboard. For example, if the entry is to be at 2532 in memory, type the following.

2532,0000	0004	0020	3103	0002.
	Channel	Equip	YC	Points to 1st
			Mnemonic	mux subtable

Bit 4 of byte 2 of the EST entry indicates to INTERCOM that this mux will be used by STIMULATOR.

The mux subtable must be set to reflect the number of terminals (this value must agree with the SIP card T parameter). Type the following.

5642,4, 00xx. xx=number of terminals

#### Running Two Multiplexers

Example 1: Two 6676s simulated on the same channel

The M parameter on the SIP statement should be set to 2. The T parameter should indicate the number of terminals on each mux. The STIMULATOR input tape must contain as many sessions as the total number of terminals: 2\*T. Two EST YC entries and two mux subtables are required. (Assemble the INTERCOM driver with IP.N6676 EQU 2.)

Example 2: One simulated 6676 and one actual 667x on another channel

It is possible to simulate one 6676 multiplexer and at the same time run another actual hardware 6676 multiplexer on a different channel. In this case, the user must ensure that an EST entry and mux subtable also exist and properly define the live multiplexer and its terminals.

The SIP parameters are set in the same way as for one simulated multiplexer.

#### Bringing up INTERCOM and the STIMULATOR

Refer also to Automatic Table Setting instructions.

Bring up INTERCOM by typing INTERCOM.

The STIMULATOR can be run at any available control point but it should be locked in to avoid attempts to roll it out. Type in the following.

n.LOCKIN.

Background batch can be run as desired. Any NEXT control point can be used to run both INTERCOM and batch jobs.

The STIMULATOR will drop automatically when INTERCOM is dropped. To drop INTERCOM type the following.

INTERCOM,DROP.

### Description of STIMULATOR Parameters

A maximum of two tapes is required for a simulation run: TAPEI and TAPEO. TAPEI is the input tape containing the Teletype programs to be simulated during the run; this tape must be assigned for all simulation runs. TAPEO is used for recording all system output resulting from the simulation; it is required if the user selects the option to recover system output. The STIMULATOR requests assignment of the appropriate tapes during its initialization phase.

The STIMULATOR is called with a Program Call control statement of the form.

SIP(Mx,Txxx,Dxxx,Sxxxx,Lxx,Cxx,Ox,Fxxxx)

Parameters M, T, D, S, L, C, O, F, and R are order-independent; all values are octal.

Mx	Number of simulated multiplexers (maximum of 2)
Txxx	Number of Teletypes per multiplexer (1-100)
Dxxx	Number of Teletypes per multiplexer to activate dynamically every S cycle
Sxxxx	Time interval in octal cycles for activating D Teletypes (one cycle equals approximately 200 ms)
Lxx	0 implies each TTY is to simulate all input programs for current run. L=0 option requires INTERCOM modifications, as same user would LOGIN at more than one terminal. 1-77 indicates the number of times each TTY is to loop on its assigned program.
Cxx	Channel number of simulated multiplexers
Ox	Zero implies recover system output, nonzero implies to bypass output
Fxxxx	File number of TAPEI to be used for current simulation
Rxxxx	Record number of TAPEI to be used for current simulation

The following default values are assumed.

M = 1  
T = 1  
D = 0  
S = 0  
L = 1  
O = 0  
F = 1  
R = 1

If the channel parameter C is not specified, SIP is terminated with an error message. The only restriction on the S parameter is the following.

$$\frac{T \times S}{D} = 4096$$

The following sample deck performs the data acquisition for a 20 terminal simulation on one multiplexer.

```
JOB1,T1000,MT02,CM20000.  
SIP(M1,T24,D24,S1,L2,CO,O0,F1)  
6/7/8/9
```

### STIMULATOR Input Statement Format

Test programs are stored on cards for input to INTERCOM via the STIMULATOR. Each card image represents one Teletype line of information. The first character must be punched in column 1 and the last character must be a v (11-0 punch).

If the input line generates a line feed as the only response from INTERCOM, as with text editing under EDITOR, the character ^ (0-8-7 punch) must precede the character v (11-0 punch) on the card. The STIMULATOR interprets the character for internal purposes only; it is sent to INTERCOM as a blank. The v character is transmitted to INTERCOM as a carriage return.

Each test program must begin with the LOGIN procedure and end with the LOGOUT system commands. The main body of the test program can contain any combination of system commands, source input, or data. Essentially, each test program represents a complete user session at a Teletype from LOGIN to LOGOUT. In converting programs from terminal input to cards for STIMULATOR input, the differences in character sets must be considered. For example, the quotations character " for terminals is the equivalence character = on cards. For a more detailed description of display and TTY characters, refer to the INTERCOM section of this document or the INTERCOM 4 Reference Manual.

The card images can be copied to tape with the COPYBF utility. EORs separate each test program.

Since the following characters have special meaning to the STIMULATOR, they should not be used as data in the input tape.

- v Indicates carriage return (11-0 punch)
- ^ Indicates EDITOR text editing line (0-8-7 punch)
- [ Indicates control X
- ] Indicates control Z

Example: ] A gives user abort

An example of input tape preparation follows.

```

JOB,MT01.
REQUEST,TAPE,HI.
COPYBF(INPUT,TAPE)
7/8/9
LOGIN.v
NAMEv      (username)
PASSWDv    (user password)
EDITOR.v
FOR,Fv
10 PROGRAM Z(INPUT,OUTPUT) v
20 PRINT 10 v
30 READ 20,A v
.
.
.
160 END v
RUN,FTNv
      2.0v
      3.0v
BYEv
LOGOUT.v
7/8/9
      (second test program)
7/8/9
.
.
.
7/8/9
      (last test program, maximum of 64)
7/8/9
6/7/8/9

```

All text editing commands require  
^ character before v character

Data Reduction Phase

The following sample performs data reduction of the STIMULATOR output tape.

```

JOB2,CM55000,T1000,MT01.
REQUEST,TAPE1,HI.    STIMULATOR OUTPUT TAPE
REWIND(TAPE1)
DATAR.
6/7/8/9

```

DATAR gives the following output.

A raw output showing number of active terminals, all response times, and a two-word debugging output for each response.

A histogram which slots all response times and calculates a mean and standard deviation. The cumulative probability column gives the probability that response time will be less than a given number of seconds. If cumulative probability is 0.4891 and the interval is 10.5-11.0, 48.9 percent of all response times were less than 11.0 seconds.

The following deck gives a more detailed report of the simulation.

```
JOB3,CM60000,T7777,MT2.
REQUEST,TAPEO,HI.    STIMULATOR OUTPUT TAPE
REWIND(TAPEO)
COPYBF(TAPEO,TAPE2)
SORT.
REQUEST,TAPE3,HI.    STIMULATOR INPUT TAPE
REWIND,TAPE1,TAPE3.
LEE(LC=77777)
7/8/9
6/7/8/9
```

LEE gives a detailed report of the activity for each terminal. For each command, the response received is shown as well as the response time in seconds and milliseconds.

The following deck does the simulation, the histogram, and the detailed terminal activity report in one job.

```
JOB4,CM70000,T1000,MT2.
SIP(M01,T05,D05,C03,S1000,O0,L1)
REWIND(TAPE1,TAPEO)
DATAR(,,TAPEO)
FILE(TAPEO,BT=K,RT=F,RB=64,FL=90,MBL=5760)
FILE(TAPE1,BT=K,RT=F,RB=64,FL=90,MBL=5760)
SORTMRG.
RETURN(TAPEO)
LEE(,,TAPE1)
7/8/9
SORT
FILE,INPUT=TAPEO(R),OUTPUT=TAPE1(R)
FIELD,FLDA(6,3,DISPLAY)
KEY,FLDA(A,COBOL6)
OPTIONS,RETAIN
END
6/7/8/9
```

#### Miscellaneous Information

Normally, the STIMULATOR will shrink its field length to the minimum required. If it is necessary to simulate a machine with less CM than is actually available (for example, 49K), the CM on the STIMULATOR Job statement can be set accordingly and FL reduction can be prevented by the following modification:

```
*IDENT,FXCORE
*DELETE,VSN.335
*DELETE,VSN.358
*COMPILE,VSM
```

The operator can examine the data captured by the STIMULATOR by displaying the output tape buffer and the TTY output pots. Suppose the STIMULATOR is running at JDT 20.

Type        C = 20

Cycle display until the FET for TAPEO is found (about 350), FIRST will be about 2040, type: C4,2040. for beginning display of TAPEO buffer.

The display can be cycled until OUTPUT pots are found. The OUTPUT pots are two 8-word buffers per TTY that give an up-to-date account of the data being received by each terminal. They are located near 3000 relative to RA.

### Type Time

The release version of the STIMULATOR has a built-in type time of 0 seconds. To change the type time, the following modification should be made to 1VG.

```
*IDENT,TYPETA
*DELETE,TYPET.1
  ADN      x   (x is the type time in octal seconds; must be ≤ 77)
*COMPILE,1VG
```

### Changes to INTERCOM

At times, changes should be made to INTERCOM, depending on the type of simulation.

When more than 30 terminals are running, the EDITOR buffers should be increased appropriately. Refer to the INTERCOM section of this document for a description of EDITOR parameters.

The appropriate version of the INTERCOM driver must be available. Refer to the INTERCOM section for more details.

### Hardware Resources Needed by the STIMULATOR

If the output is being saved on tape, the STIMULATOR requires two dedicated PPs; but only one is required if output is not saved.

A free data channel is required for communication between the STIMULATOR and the INTERCOM drivers.

The STIMULATOR uses one control point; the field length depends on the length of the input tape and the number of simulated terminals. (Approximately 25K [octal] for a typical run consisting of one mux and an input tape consisting of 64 sessions, 30 lines per session.)

### SIS63 USAGE

The following control statement is required for execution.

```
SIS63(lfn)
```

where lfn is the SIS file name.

RELEASE MATERIALS

FORTRAN Extended 4 is released on one reel of tape, PL7, which contains the compiler. PL7A is used by those installations who have purchased the single-pass time-sharing (TS) version of FORTRAN Extended. The installation of FCL 4 (PL8) mathematical and I/O libraries is required for FORTRAN Extended 4 execution (refer to section 34).

The structure of PL7 or PL7A is the following.

File 1	Program library of the FORTRAN Extended 4 compiler
File 2	Relocatable binary
File 3	Relocatable binary
File 4	Absolute overlay binary

LIMITATIONS

All applicable Integer Multiply FCOs must be installed. All code generated by the compiler assumes the existence of the Integer Multiply.

If FORTRAN Extended 4 is installed on a CYBER 70 model 71, 72, or 73, or a CYBER 170 model 171, 172, 173, 174, 720, or 730 with the MODEL installation parameter (in IPARAMS) correspondingly set, the object code produced will execute properly but will not be optimal for a model 74, 175, 750, or 760. If MODEL is set to 74 or 175, the object code produced will execute properly on a model 71, 72, 73, 74, 171, 172, 173, 174, 175, 720, 730, 750, or 760, but will be optimal only for the model selected. If the MODEL parameter is set to 176, the compiled object code will not execute correctly on other models when the source programs contain LEVEL 2 (direct access LCM) statements, but will execute correctly although not optimally on other models when the source programs do not contain LEVEL 2 statements.

Object code compiled by FORTRAN Extended 4 on a model 71, 72, 73, 74, 171, 172, 173, 174, 720, or 730, cannot be executed on a Model 76 running under the SCOPE 2 operating system. On the lower CYBER models, object code consists of one logical record for each program unit, whereas the SCOPE 2 operating system loader accepts only W records.

When the FTN control statement specifies either the C or E option, the compiled object code is produced as symbolic COMPASS source language, rather than executable binary. The C and E options may not be selected in TS mode.

COMPILER INSTALLATION PARAMETERS

The FORTRAN Extended 4 compiler program library, PL7 or PL7A, is distributed with installation parameters properly set for normal installation on any CYBER 70 or CYBER 170 series machine. It should be noted that the system text IPTEXT should contain parameter values which are consistent with the CYBER model on which the compiler is installed and executed. To ensure correct code generation, the MODEL micro in deck IPARAMS on PL1A must be set to the correct value for the target machine.

The installation options are located in the common deck OPTIONS and deck FTN. OPTIONS is called by TSTEXT, FTNMAC, and FTNTEXT; because of its global nature, the compiler should be reinstalled whenever parameters are changed. Installation parameters in FTN may be revised through a standard maintenance run (installation deck PL7I).

Current UPDATE sequence numbers for installation options may be obtained by assembling FTNMAC (or TSTEXT, FTNTEXT; the FTNMAC listing is much shorter) and/or FTN, depending on the parameters of interest. FTN contains the installation parameters for default control statement settings, control statement error processing, default file names, object-time input/output buffer length, and compiler overlay library names. The remaining parameters are in OPTIONS (FTNMAC/FTNTEXT/TSTEXT).

The default external and internal file names used by the compiler include the following.

INPUT	Source program input
OUTPUT	Compiler listable output
LGO	Relocatable binary object code
COMPS	COMPASS formatted symbolic object code (E option only)
ZZZZFC	Internal symbolic object code (E option not selected)
ZZZZRL	Internal intermediate language
ZZZZRM	Internal reference map
ZZZZOP	Internal OPT=2 and DEBUG-mode random file

All files are formatted according to suitable operating system standards. File formats cannot be changed through FILE control statements. (CDC CYBER Record Manager has not been implemented for compile-time I/O. The upper CYBER implementation of Record Manager has been designed only for standard file formats; results are unpredictable if FILE statements are used.)

### COMPILER PROGRAM LIBRARY STRUCTURE

When a full UPDATE is performed on PL7 or PL7A, the following eight records are written on the compile file.

	<u>Contents</u>	<u>Overlay</u>	<u>Program Library Deck Names</u>
1.	TS mode global assembly text (used only for installation or maintenance)		TSTEXT
2.	Object code macro definition text		FTNMAC
3.	Two-pass global assembly text (used only during installation or maintenance)		FTNTEXT
4.	Master controller	(0,0)	FTN
5.	TS option one-pass compiler. This record is empty if the compile file was produced from PL7, the non-time-sharing compiler.	(1,0)	FTN10-INIT
6.	Two-pass compiler (OPT=0, 1, or 2) batch compilation controller	(2,0)	FTN20-SNAP
	Error message text expander	(2,3)	FTN23-FTNMSG
	Pass 1 (non-DEBUG)	(2,1)	FTN21-PHI CTL
	Reference map processor and assembler	(2,5)	FTN25-REFMAP
	Pass 2	(2,2)	FTN22-MACROX
7.	Pass 1 DEBUG code supplement	(2,4)A	FTN24-SAVREGS
8.	Pass 1 skeleton COPYL decks	(2,4)B	FORMDBG

#### TSTEXT

TSTEXT is a global set of macro, micro, and symbol definitions needed to assemble the (1,0) overlay. When the compiler is installed and maintained, TSTEXT is first assembled as a local text file and then accessed by the COMPASS or FORTRAN Extended 4 G option parameter.

#### FTNMAC

FTNMAC is a system text that contains macro definitions needed to assemble symbolic object code compiled by FORTRAN Extended 4. Normally, FORTRAN Extended 4 produces executable binary (not symbolic) object code without using FTNMAC. However, if the C or E option is selected, FORTRAN Extended 4 produces object code in a symbolic form for COMPASS assembly. The symbolic code contains many macro calls which must be externally defined for successful assembly. The macro definitions are available in FTNMAC, which is assembled and added to the operating system nucleus library during compiler installation.

FTNMAC must be assembled with the same version of COMPASS as is present when FORTRAN Extended 4 is eventually added to the running system.

#### FTNTEXT

FTNTEXT is a global set of macro, micro, and symbol definitions needed to assemble the (0,0) and (2,n) overlays of FORTRAN Extended 4. When the compiler is installed and maintained, FTNTEXT is first assembled as a local text file, and then is accessed by the COMPASS or FORTRAN Extended 4 G option parameter.



## (0,0) Overlay (FTN)

This overlay is the master controller which does the following.

- Scans, validates, and stores FORTRAN Extended 4 control statement option parameters.
- Initializes the compiler according to control card options and available memory.
- Loads all compiler primary and secondary overlays.
- Processes all operating system action requests.
- Loads and communicates between the COMPASS (1,0) overlay and FORTRAN Extended 4 for intermixed COMPASS language program units.

## TS MODE OVERLAY

### (1,0) Overlay (FTN10)

This overlay contains the entire TS mode compiler (except for the (0,0) controller) and remains resident in core for the entire compilation. Source input is read, listed (if requested), and used to generate code. If COMPASS subprograms occur on the input file, COMPASS is loaded to assemble them and FTN10 is reloaded, if necessary, for subsequent FORTRAN subprograms.

## OPTIMIZING AND DEBUGGING MODE OVERLAYS

### (2,0) Overlay (FTN20)

This overlay is the batch controller for compiling multiple FORTRAN program units. It contains the symbol/label table lookup subroutines; the central compiler input/output subroutines; the batch compilation reinitialization code; miscellaneous utility subroutines; and the compiler malfunction report package.

### (2,3) Overlay (FTN23)

This overlay expands two-word error table entries into full line error messages. It is loaded only if errors were detected during Pass 1 of a compilation.

### (2,1) Overlay (FTN21)

This overlay is the first pass of the compiler under normal mode (when the DEBUG option is not selected by the control statement D parameter). It performs a lexical, syntactic, and semantic analysis of each FORTRAN program unit. Source language input is translated through a lexical element language (E-list) to a register-independent internal language (R-list). Source language errors are detected and saved in an error table for subsequent expansion. Intermixed COMPASS language programs are recognized and either copied to an internal file or transmitted directly to COMPASS, depending on control card option selection.

### (2,2) Overlay (FTN22)

This overlay is the second pass of the compiler. It optimizes and generates symbolic object code from the R-list produced by Pass 1.

### (2,5) Overlay (FTN25)

This overlay is the third pass of the compiler. A reference map is produced, if requested. The symbolic code is then assembled as executable binary object code, either by a fast one-pass internal assembler or (if the C option is selected) by the slower COMPASS assembler. This overlay can be combined with the (2, 2) overlay during installation, as selected by the symbol .OVL in OPTIONS. Compiler loading time is reduced, but compiler field length must be increased.

### (2,4) Overlay (FTN24)

This overlay is loaded only when the DEBUG option is selected on the FTN control statement. It is loaded instead of the normal Pass 1; it contains, in addition to all normal Pass 1 code, processing subroutines for DEBUG statements. The overlay is formed during installation by assembling the DEBUG subroutines and then replicating the normal Pass 1 code with the COPYL utility.

Minimum DEBUG field length is 63K (octal) or approximately OPT=0+15K.

## INSTALLATION INSTRUCTIONS

The compiler installation decks provide a method for introducing the FORTRAN Extended 4 compiler into a NOS/BE system. The first job PL7I updates the program library, producing a new program library tape including supplemental binary files. Deck PL7E must be run following PL7I but before attempting installation of the object library when the running system modification approach to building systems is used.

Deck PL7I references IPTEXT and CPUTEXT; part III of this document contains a cross-reference map of referencing routines versus IPARAMS symbols. Deck PL7I also requires access to the COMPASS program library to acquire the common deck COMPCOM.

Compiler installation job decks PL7I and PL7E, and verification program PL7V can be obtained from the Installation Deck program library using the procedure outlined in part I, section 1.

Decks PL7E and DST3 need not be run if the user library approach is being followed.

RELEASE MATERIALS

COBOL Version 4 release material consists of a magnetic tape identified as PL9. The structure of PL9 is as follows.

- File 1 UPDATE version of the COBOL program library
- File 2 COBERTX COBOL error text
- File 3 Relocatable binary records resulting from assembly of the object time routines
- File 4 Relocatable binary records resulting from assembly of the compiler routines
- File 5 COPYCL binary decks
- File 6 Overlays forming the COBOL compiler installed into the system
- File 7 COPYCL routine in absolute form

LIMITATIONS

Integer multiply hardware FCOs are required. Binary decks of COBOL source programs produced with an IPARAMS symbol value of IP.IMUL CEQU 0 are now invalid: the source deck must be recompiled under NOS/BE. No source deck modifications are needed.

The compiler may be executed from any file or user library providing the main overlay and all other overlays reside on the same file or library; or the main overlay resides in NUCLEUS and the other overlays are in SYSOVL.

INSTALLATION PARAMETERS

The COBOL compiler uses symbol definitions from IPTEXT for IP.CMU, IP.TYPE, IP.CSET, IP.PS, and IP.PD (refer to section 1 discussion of IPARAMS). To override these installation parameter values, make the following changes in the COMDECK ASSEMOP when COBOL is assembled.

<u>Feature</u>	<u>Required Change</u>
Generate code optimized for a Model 74 type CPU	*B,ASSEMOP.4 SKIP 1
Generate code optimized for a Model 73 type CPU	*I,ASSEMOP.4 SXFR EQU 1
Generate CMU instructions	*D,ASSEMOP.9,10
Generate non-CMU instructions	*D,ASSEMOP.9
Specify COBOL output page density (lines per inch) prior to the addition of IP.PD to IPTEXT	*D FEAT172.11 IP.PD EQU n (where n can be 3, 4, 6, or 8)
Specify COBOL output page size (lines per page) prior to the addition of IP.PS to IPTEXT	*D FEAT172.14 IP.PS EQU n (where n can range from 4 through 99999)

Part III contains a cross mapping of referencing routines and IPARAMS symbols defined in IPTEXT.

SPECIAL INSTALLATION PROCEDURES

To create a compiler and an object library which contain the CDCS 1 interface feature activated, make the following change in the COMDECK ASSEMOP when the compiler and object library are assembled.

```
*I ASSEMOP.23
DB1.1 EQU 1
```

## INSTALLATION PROCEDURES

The release tape, PL9, contains seven files. File one contains the COBOL program library. This file includes both compiler and object routines. Files 2 through 7 contain binary decks of the object time routines and the compiler overlays.

Installation job decks PL9I, PL9E, PL9O, and PL9V may be obtained from the Installation Deck program library using the procedure outlined in part I, section 1.

Job PL9I uses the release tape as input to create a tape of the same structure.

When job PL9E is performed, using either the released tape or the output tape created by job PL9I, COBOL is added to the running system or user libraries by EDITLIB. Job DST3 can be used to generate a deadstart tape. Jobs PL9E and DST3 need not be run if the user library installation approach is being used.

Deck PL9O allows regeneration and replacement of absolute overlays on a new PL9 tape.

## RELEASE MATERIALS

Sort/Merge 4 is released on release tape PL10. PL10 contains the following files.

<u>File</u>	<u>Contents</u>
1	Sort/Merge program library
2	Sort/Merge relocatable binary decks (macro sort)
3	Sort/Merge (0,0) overlay
4	Sort/Merge capsule relocatable binaries
5	SMTEXT binary

## HARDWARE CONFIGURATION

Sort/Merge requires the same minimum hardware configuration as NOS/BE. If the Tape Sort option is used, additional magnetic tape units are required: polyphase requires at least three; balanced requires at least four.

## GENERAL DESCRIPTION

The system consists of two directive formats and a Macro Sort format. Directive format 1 is based on upward compatibility toward 7000 Sort/Merge and Format 2 is based on the Sort/Merge 3 format. The Macro Sort format is also based on 7000 compatibility. Sort/Merge 4 is a more modular package, consisting of capsule modules which are in core only when necessary. For example, a disk sort does not need the tape merge capsule modules. This product is designed to optimize speed and core space as well as to utilize CDC CYBER Record Manager and NOS/BE capabilities.

## INSTALLATION PARAMETERS

Sort/Merge selects installation parameters from IPTEXT. Sort/Merge selects CMU versus non-CMU use dynamically at execution time.

## INSTALLATION PROCEDURE

Job decks PL10I, PL10E, PL10O, PL10V1, PL10V2, and PL10V3 may be obtained from the Installation Deck program library using the procedure outlined in part I, section 1 of this document.

Job PL10I references IPTEXT; Part III of this document contains a cross reference map of IPARAMS symbols versus referencing routines.

The installation jobs function as follows.

- PL10I      Updates the program library with modifications producing a new program library tape including assembled binary information as supplemental files. This job essentially allows creation of a revised release tape.
- PL10E      Adds Sort/Merge to the running system or user libraries. PL10E can use either the released PL10 or a tape created by job PL10I as input.

Deck PL10O, applicable only in the user library method of installation, allows regeneration and replacement of absolute overlays in the user library plus creation of a new PL10 tape.

After job PL10E has been run, job DST3 can be run to create a deadstart tape of the running system containing Sort/Merge. Decks PL10E and DST3 need not be run if the user library installation approach is being used.

The system text SMTEXT is installed via execution of installation decks PL10I and PL10E.



RELEASE DESCRIPTION

INTERCOM 4 in conjunction with the NOS/BE operating system provides TTY and CRT terminals with time-shared access to CYBER 170, CYBER 70, and 6000 Series computers. Also, remote batch jobs can be submitted from terminals equipped with a remote card reader and printer, from a remote computer (1700 or 8231) running an IMPORT package, or from a low-, medium-, or high-speed batch terminal. Programs written in the FORTRAN, COBOL, ALGOL, COMPASS, or BASIC languages can be submitted from a remote terminal for execution at control points; the user at the remote terminal can interact with the executing program. Program output can be routed to the line printer and card punch at the central site or to a terminal equipped with line printers or card punches. Through the system permanent file feature, input from a central site magnetic tape or card reader is available to the remote user.

HARDWARE CONFIGURATION

In addition to the minimum hardware required by the NOS/BE system, INTERCOM 4 requires the following equipment for communication and operation.

- A CRT terminal, model 214-11, 214-12, 217-11, 217-12, 217-13, 217-14, 711-10, 731-12, 732-12, 734 Remote Batch Terminal, 714-10, 714-20, or CDC CYBER 18, or a model 33, 35, or 38 KSR or ASR Teletype terminal, or a 713 Teletype compatible terminal, or a 751 TTY compatible terminal, or a 1700 or 8231 remote computer running an IMPORT package, or a 731-10 Low Speed Batch Terminal (LSBT), 732-10 Medium Speed Batch Terminal (MSBT), or a 733-10 High Speed Batch Terminal (HSBT).
- A dedicated multiplexer on a dedicated channel 6671, 6676, 6673, or 6674 (6671 for Teletype and/or CRT terminals; 6676 for Teletype terminals only; 6673 or 6674 for high speed connections to remote computer); a dedicated 7077 Communications Station on a dedicated channel with a 791 Local Communications Controller (LCC) for Teletypes, XSBTs, and/or CRTs; a dedicated 2550 Network Processing Unit (NPU) on a dedicated channel for TTY and/or CRT terminals.
- Data sets for communication between the remote terminal and central site. Teletype terminals require 103A or 212A Data Sets; CRT terminals require 201A or 201B Data Sets, or CDC 358 Transceivers; remote computer and HSBTs require 301B or 303 Data Sets and a TELPAK A communication line, or CDC 358 Transceivers. Refer to the Control Data Communications Handbook for specific details of the exact modem strapping option required by INTERCOM 4.

REQUIRED HARDWARE OPTIONS

711-10		
Data control	711-102	
714-10 or 714-20		
Display (8x80 or 16x80)		714-122 or 714-123
733-10		
Data set adapter	733-130	
CRT (16x80 or 18x64)		733-150 or 733-152
731-12		
732-12		
Memory increment (8K bytes)		730-100
Display (16x80)	730-101	
731-10		
732-10		
Display (16x80)	730-101	
Cyclic encoder	730-103	

HARDWARE OPTIONS

Teletype		
Paper tape reader/punch		
217-11, 217-12, 217-13, 217-14		
Card reader 224-11, 12, 13, or 14		
Line printer 222-11, 12, 13, or 14		
711-10		
Memory option (16x80 screen)	711-100	
Character printer	711-120 or 711-21	

714-10  
 Display (8x80 or 16x80) (up to 8 additional) 714-122 or 714-123  
 Character printer (up to 3) 711-120 or 711-121

731-10

732-10  
 Card punch/reader (66/330 cpm) 730-104  
 Memory increment (if additional peripherals are used beyond the basic  
 L/MSBT card reader and line printer) 730-100

733-10  
 Card reader (one additional) 733-120  
 Line printer (up to three additional) 733-110  
 Card punch 733-101  
 Memory increment (if additional peripherals are used beyond the basic  
 HSBT, single card reader and line printer) 733-140

Refer to the Software Release Bulletin for the controlware part numbers for which INTERCOM has been tested.

#### RELEASE MATERIALS

INTERCOM Version 4 release material consists of a magnetic tape (PL12) containing the INTERCOM program library as file one.

#### NOTES AND CAUTIONS

##### MODE 4 SEQUENCE BIT PROCESSING IN 1M1

For synchronous mode 4 terminals, the INTERCOM 1M1 driver checks the sequence bit on the station address word of responses to all display mode transmission.

Some mode 4 CRT terminals work properly in all respects except that they do not correctly process the sequence bit; generally, the terminal sends a zero sequence bit in all transmissions to the 6000 computer. This problem is a terminal malfunction, a loose wire, or bad hardware card. Such terminals may undergo endless retransmissions of one of the first two WRITES to the CRT screen. Should this retransmission be observed regularly when a particular terminal connects, hardware support personnel should check the sequence bit (bit 24) in the station address word.

##### LINE SKIPPING ON 714 NONIMPACT PRINTER

When column 80 is reached, an INTERCOM-generated line skip occurs. 714 nonimpact printers have a photo cell switch set at column 80 to skip a line. Thus, when input lines exceed 79 characters, output is double-spaced. If this is not desirable, request the site customer engineer to move the photo cell switch beyond column 80.

##### ERROR PROCESSING ON THE 711/714

On the 714, the ERR enable/disable switch should be set to ENABLE. On the 711, the ERR disable jumper should not be present. This allows the controller to give an ERR response both to writes to nonexistent stations, and to invalid messages in general.

##### CHARACTER SET SUPPORT

The IP.CSET Display Code character set selection affects INTERCOM only with respect to 63- or 64-character set selection. The BCD and ASCII printer character set default and 026/029 keypunch code default selections affect only the remote batch terminals. How it affects each one is described in each terminal's reference manual.

In addition to the IP.CSET display code character set selected, INTERCOM allows users to select extended ASCII 95- or 256-character sets for communication with a mode 3 type terminal. These sets are described in the INTERCOM Reference Manual.

##### 2XPPU SPEED (CDC CYBER 170 ONLY)

When INTERCOM is run at 2XPPU speed (IP.PPS2X=2) with a 6676 multiplexer, model F 6676 must be used.



## LIMITATIONS AND SYSTEM CONSIDERATIONS

1. When the CONNECT command (or CONNEC call) is used, the specified data is routed to or from the terminal each time the file is read or written. When simultaneous operations are to be performed, no more than one file should be connected to a terminal for interactive operations at any time.
2. PASSWRD should not be run while INTERCOM is up because numerous problems may occur if a user id is changed while INTERCOM is up. These include remote batch file security problems and EDITOR's edit files being swapped between users.
3. Teletype operation via the 791 LCC differs from Teletype operation via the 6671 or 6676 multiplexer.

8-bit data is not currently supported on mode 3 (TTY) devices connected to the LCC. Improperly formatted paper tape input to the LCC may cause the TTY to be inoperative.

## GENERAL PROCEDURES

Installation of a complete INTERCOM system requires establishing installation parameters and installing from the INTERCOM OLDPL. The card deck described later can be run at the central site to install INTERCOM. FORTRAN Extended 4 and COMPASS must be installed before INTERCOM can be installed.

## INSTALLATION PARAMETERS

Configure the INTERCOM system for a particular installation by performing the following.

- Parameters in the INTERCOM common deck INTCOM may be changed to affect the characteristics of INTERCOM.
- An equipment status table (EST) entry must be established for each multiplexer dedicated to INTERCOM.
- In CMR, a multiplexer table must be defined which contains subtables for each multiplexer dedicated to INTERCOM.
- The installation deck must contain an assembly for each variant of the low speed multiplexer driver required.
- Certain tables within 1CI, 1QP, and 3TT can be set to control use of selected commands.
- Parameters in the EDITOR common decks IPFTN and IPCOM may be changed to affect the characteristics of EDITOR.
- Parameters in the multiuser job common decks MUJCOM and CMUJCOM may be changed to affect the characteristics of multiuser jobs (particularly EDITOR).

## INTERCOM COMMON DECK SETTINGS

Release values are shown in the following list of INTERCOM parameters for the common deck INTCOM present on PL12. If these parameters are to be changed, the cards containing the proper code with the CEQU macro should be placed after an \*INSERT INTCOM.43 directive and inserted into the first update record of the deck PL12I. Alternate tested values are shown in parentheses.

A cross-reference listing showing the routines that reference each INTCOM and IGSCOM symbol may be found in Part III.

IP.IACES CEQU 11

An 11-bit field contains the user table access field and user permission bits. This value must be the same as the value for IP.IACES (refer to NOS/BE IPARAMS in this section). The entire 11-bit field is used to determine if a user has access to a specific utility or routine. The setting of IP.IACES determines how many bits, right-justified are to be used as the access level. The remaining bits (11-IP.IACES) are used as permission bits.

User access level is an octal integer (range 0 to (2\*\* IP.IACES)-1) and is contained in the user table after the user logs in. User's access level must be greater than or equal to the command's access level in order to use a command.

Permission bits form a mask constant (range 0 to  $(2^{11-IP.IACES})-1$ ). Each bit which is set in the command's permission-bit mask must also be set in the user's permission-bit mask in order for the user to use the command.

IAJ and LOADER check permission bits and access levels for commands found in the NUCLEUS Entry Point Name Table.

A program in a library, specifically the Entry Point Name Table entry in the NUCLEUS library, has an 11-bit permission bits/access level value. In addition, only this type of command verb has one additional bit associated with it indicating whether the entry is control-statement-callable. In the EPNT entry, bits 14-4 contain the permission bits and access level required; bit 3 contains the control-statement-callable bit (0 = not control statement-callable). IAJ checks bit 3 for all control statements.

EDITLIB allows definition of permission bits and access levels via the SETAL directive or the AL parameter of the ADD and REPLACE directives. This value is not access level; it is a 12-bit value combining permission bits (upper 11-IP.IACES bits), access level (bits IP.IACES-1), and control-statement-callable (bit 0). The upper 11 bits of this value are the required permissions and access level found in bits 14-4 of the EPNT entry.

During a PASSWRD run, a user's permissions and access level are defined via the A=accevl parameter. This value is an 11-bit octal number combining permission bits and access level. No control statement-callable value is associated with the user's accevl value.

IP.IACES may be given any value between 0 and 11. If IP.IACES = 0, then the entire field is permission bits. If IP.IACES = 11, then the entire field is access level.

**Example**

1. IP.IACES = 6
2. EDITLIB run with directives  
     SETAL (FILES, 201)  
     SETAL (ASSETS, 407)
3. PASSWRD run with directives  
     ADD U=USER1, P=PASS1, A=2  
     ADD U=USER2, P=PASS2, A=302  
     ADD U=USER3, P=PASS3, A=3077  
     ADD U=USER4, P=PASS4, A=1515  
     ADD U=USER5, P=PASS5, A=0712
4. As the result of the preceding installation, the following relationships exist.

<u>COMMAND</u>	<u>PERMISSIONS ASK</u>	<u>ACCESS LEVEL</u>	<u>PERMISSIONS REQUIRED</u>
FILES	1	0	0
ASSETS	2	3	1

<u>USER</u>	<u>PERMISSIONS ASK</u>	<u>ACCESS LEVEL</u>	<u>PERMISSIONS GRANTED</u>
USER1	0	2	NONE
USER2	3	2	0,1
USER3	30B	77B	3,4
USER4	15B	15B	0,2,3
USER5	7	12B	0,1,2

	<u>FILES</u>	<u>ASSETS</u>	
USER1	P0	P1, AL	Pn Denied because user lacks permission n
USER2	ALLOWED	AL	AL Denied because user access level too low
USER3	P0	P1	
USER4	ALLOWED	P1	
USER5	ALLOWED	ALLOWED	

**USER X COMMANDS ALLOWED**

IP.IBBMN CEQU 24

Minimum number of (small) buffers needed once remote batch activity has been initiated.

IP.IBBMX CEQU 36

Maximum number of (small) empty buffers needed once remote batch activity has been initiated.

IP.ID CEQU 1

If one, the INTERCOM user id is used as the default permanent file id by commands STORE, FETCH, and DISCARD. If zero, the permanent file id must be specified by the INTERCOM user.

IP.IDFL CEQU 55000B

Default field length assigned to a user's program when the user has not entered a field length (EFL).

IP.IDFLE CEQU 0

Default ECS field length (in multiples of 1000B) allowed to a user if no E parameter was specified for the user's ID on the password file.

IP.IFC CEQU 700 MILLISECONDS OF DELAY FOR FIRST CHARACTER

When the IM1 driver completes a transmission to a CRT device, it assumes that the terminal will receive the message, that the line will turn around, and that the 6000 will receive the first input character from the terminal within IP.IFC milliseconds. The driver will consider the terminal to have not responded (to have timed-out on the transmission) if no response of any type is detected within this interval. About 700 milliseconds should be adequate for most installations.

IP.IGCON CEQU 0

Maximum number of 1700/274 Graphics console entries in Console Reservation Table; should be set to zero if Graphics is not defined in the system. IP.IGCON has a maximum possible value of 24.

IP.IGS CEQU 0 (1)

If one, 1700/274 Graphics is defined as existing within the system; if zero, 1700/274 Graphics is not present in the system.

IP.IHEAD CMICRO 0, (CONTROL DATA INTERCOM 4.7)

Header output by IIM when a remote terminal dials into the INTERCOM system.

IP.IBMN CEQU 12

Minimum number of (small) empty buffers needed when no remote batch activity has been initiated.

IP.IBMX CEQU 18

Maximum number of (small) empty buffers needed when no remote batch activity has been initiated.

IP.IM2DW CEQU 80D

Default screen width in characters for mode 2 terminals (refer to Port Definition Entries for mode 2 terminal list). This value should be  $10D \leq x \leq 80D$ .

IP.IM2DS CEQU 1120D

Default screen size in characters for mode 2 terminals (80x14). This value should be  $20D \leq x \leq 2047D$ .

IP.IM3DW CEQU 72D

Default line length in characters for mode 3 terminals (refer to Port Definition Entries for mode 3 terminal list). This value should be  $10D \leq x \leq 132D$ .

IP.IM3DS CEQU 4095D

Default page size in characters for mode 3 terminals. This value should be  $20D \leq x \leq 4095D$ .

IP.IM4DW CEQU 50D

Default screen width in characters for mode 4 terminals (refer to Port Definition Entries for mode 4 terminal list). This value should be  $10D \leq x \leq 80D$ .

IP.IM4DS CEQU 1000D

Default screen size in characters for mode 4 terminals (50x20). This value should be  $20D \leq x \leq 1280D$ .

IP.IND CEQU 3

Maximum number of active INTERCOM drivers (of any type) allowed in the system simultaneously. It should never exceed six.

IP.IPRLS CEQU 100B

Priority loss per 100g PRUs used by an output file after the first 100g PRUs (refer to the IP.MPRIT description).

IP.ISFL CEQU 2500B

Default swap-in field length for INTERCOM. The swap-in field length is the amount of memory requested to swap in an INTERCOM command.

IP.IWT CEQU 7B TIMED-OUT PORT DELAY

When the 1M1 driver has timed-out without getting any response from a hardwired or dedicated mode 4 (200 UT protocol) terminal a number of times consecutively, it assumes that the terminal is inoperative; this means either powered-down, broken, or nonexistent, or that somehow the communication link with that terminal is down. In this situation, the driver does not waste processing time trying to communicate with the terminal but marks it timed-out and attempts to communicate with it periodically on the assumption that it may come up at any time. The frequency with which the driver attempts communication with timed-out ports is proportional to IP.IWT and is usually on the order of once every few minutes.

At a computer site where the communication lines are not reliable and where operative terminals time-out frequently, IP.IWT should be set to 1 (one) or in extreme cases 0 (zero) to allow 1M1 to test timed-out ports more frequently. Unless good ports are frequently not being polled due to being timed-out, it is recommended that this parameter be unaltered.

IP.IMXL3 EQU 700

Maximum number of data characters which can be contained in a single Teletype data transmission from the 2550 Front-End to the central site system. IP.IMXL3 must be greater than or equal to the TTY TIP parameter /TTYIBL described in section 23. This parameter must not exceed 2043.

IP.IMXL4 EQU 1280

Identical to the preceding IP.IMXL3 description except it applies only to mode 4 transmission; the relevant mode 4 TIP parameter is /MD4IBL.

IP.LP4C CEQU 1

Defines the type of mode 4C printer in use. 0 = impact printer, 1 = nonimpact printer.

IP.MALOC CEQU 4000B

A 12-bit octal value defining the allocation style for files created by a multiuser job. Bit 11 always is set to one to indicate that a permanent file device is requested. The bits indicating the allocation style are bits 5 through 0. This value is placed in the File Name Table entry generated for new multiuser job files, in byte C.FALLOC.

IP.MPRIT CEQU 4000B

Maximum priority to be assigned to an output file diverted by INTERCOM. If fl is the length of the file in PRUs, the priority assigned to a file can be expressed as  $IP.MPRIT - (IP.IPRLS * (fl - 100g) / 100g)$ , where / denotes an integer divide.

IP.MXCOR CEQU 2500B

Maximum field length allowed for INTERCOM buffer use (in multiples of 100 octal words); cannot exceed 400 000g.

IP.PRIX CEQU 3777B (7000B)

Nonzero indicates the priority given to input files read from remote site. If zero, priority will be taken from job statement.

IP.SNIFL CEQU 22000B

Initial field length for a SIGNON job. IP.SNIFL is an invariant parameter.

IP.SNOTL CEQU 600

Time limit for jobs entering system through 274IGS.

IP.SNOFL CEQU 60000B

If nonzero, maximum field length (RFL or MEM) allowed for a job entering the system through 274IGS.

IP.TSL CEQU 10B

Default time limit in seconds for execution of a user's program, if the user has not entered a time limit (ETL).

IP.1LX CEQU 1

If 0, no remote batch is allowed. No CM buffers are reserved.

If 1, one copy of 1LX is called to support all remote batch terminals. One CM buffer is reserved.

When IP.1LX is greater than one, it must equal the number of copies of 1LX that can be active. This number of CM buffers will be reserved. One copy of 1LX is called for each 6671 driver with batch terminals, and one copy for each LCC and 2550 equipment with batch terminals.

#### NOS/BE IPARAMS SETTINGS

These parameters must be set at \*INSERT IPARAMS.15 when NOS/BE is installed (deck PL1A1).

IP.IACES CEQU 11D

Defines the number of bits in the access level, for use by 1AJ and LOADER. This value must be the same as that specified for the INTERCOM parameter IP.IACES.

IP.ILCMD CEQU 1

If set to 1, the last word in the user table will store the last command entered by each user for display on the DSD Q display. If 0, it will not be used for this purpose.

IP.IUSID CEQU 2RAJ

Defines the first user id available for assignment by the program PASSWRD. The value of this parameter is determined by the number of high-speed multiplexers with subtables defined in the system and the number of hardwired remote batch or mode 2 terminals defined in the system. The high-speed multiplexers use two ids per 6673 or 4 ids per 6674, starting with user id AJ. The hardwired remote batch terminals and mode 2 terminals use one id per terminal.

This user id is the lowest available to be assigned an interactive user. Every remote high speed batch terminal connected to the system must have its own terminal id assigned to it.

IP.1M1 CEQU 1

Should be zero if the system has no 6671 or 6676 multiplexers; otherwise, should be nonzero.

IP.1WB CEQU 1

Should be zero if the system has no 6673 or 6674 multiplexers; otherwise, should be nonzero.

IP.1ZZ CEQU 0

Should be zero if system has no LCCs; otherwise, should be nonzero.

A cross mapping of referencing routines and all symbols in IPARAMS (IPTEXT) can be found in part III.

#### EST ENTRY

The EST table, established when deck PL1A1 is run to install NOS/BE must contain an entry for each multiplexer dedicated to INTERCOM. The channel referenced in this entry must be dedicated to the INTERCOM multiplexers on that channel. For nonallocatable equipment, the EST uses the EST macro which has been modified as follows.

type EST parameters (of the form key = value)

Macro parameters used by INTERCOM include the following.

type DC for 6671, YC for 6676, SC for 6673 or 6674, CS for 791, FE for 2550

CH= Channel for multiplexer or 7077 Communication Station or 2550 Front End

EQP= Equipment number for multiplexer or 7077 SAC/CSM I/O channel for 791 or 2550 Front End

MOD= OFF if off, otherwise do not use

MUX= Index to INTERCOM multiplexer table

A typical EST entry might appear as follows.

```
*I EST.1
DC EST CH=3,EQP=5,MUX=MUX1-T.ITABL
```

This entry notifies the multiplexer driver that a 6671 with equipment number 5 is on channel 3; and the index to the multiplexer subtable for this 6671, and T.ITABL is the beginning of the multiplexer table.

Typical EST entries for two LCC 791s on SAC/CSM channels 0 and 1 connected to a 7077 connected to channel 4 would appear as follows.

```
CS EST CH=4,EQP=0,MUX=MUX1-T.ITABL
CS EST CH=4,EQP=1,MUX=MUX2-T.ITABL
```

#### CONFIGURATION PARAMETERS (INTERNAL TO CMR)

This parameter defines the length of the INTERCOM multiplexer table. It must be set at \*INSERT CMRIP.1 when NOS/BE is installed. The default value is the following.

```
L.ITABL CEQU 19
```

This parameter should be changed to reflect the size of the multiplexer table for each installation. The length of the table can be determined from the following formula.

```
L.ITABL = 2+2*N76+N71+N71 PORTS+2*N73+2*N74+2*N91+N91 PORTS+N50+N50PORTS
```

N76	Number of 6676 multiplexers dedicated to INTERCOM
N71	Number of 6671 multiplexers dedicated to INTERCOM
N71 PORTS	Total number of 6671 ports defined
N73	Number of 6673 multiplexers dedicated to INTERCOM
N74	Number of 6674 multiplexers dedicated to INTERCOM
N91	Number of 791s
N91 PORTS	Total number of 791 ports defined
N50	Number of 2550s
N50 PORTS	Total number of 2550 ports defined

#### CMR MULTIPLEXER TABLE

The CM resident INTERCOM multiplexer table is used by INTERCOM to provide data on the hardware configuration of the installation and to record parameters. It consists of two dedicated parameter words and one or more subtables assigned to the multiplexers serviced by INTERCOM.

The first two words of the multiplexer table, the parameter words, start at location T.ITABL in CMR and are already assembled into CMR. The subtables follow the parameter words in any order convenient to the installation. The first subtable must be defined at \*INSERT MUX.1 when NOS/BE is installed. Each subtable has a relative pointer in the EST entry for that multiplexer. The upper bound of the multiplexer subtable may not extend beyond 7777B.

For the 2550, Port 0 must be empty and baud rates must be specified in descending order for ports.

#### CMR MULTIPLEXER SUBTABLE GENERAL FORMAT

Each multiplexer subtable contains one macro to define the type of multiplexer, followed by one macro for each port defined on that multiplexer if a 6671, 2550, or a 791. The address of the macro describing the multiplexer is the same address used in the EST entry defining that multiplexer. A subtable for a 6671 multiplexer might be defined as follows.

```
MUX1      MUX71      4
           TMB4A
           TMA4A
           TM3
           TM3
```

When a 6671 multiplexer or 2550 Front End is configured, it is advisable to place the highest speed terminals on the lowest ports and to place any empty ports at the high number port positions. Thus, the 6671 and 2550 should be configured 9600 baud terminals first, then 4800 baud terminals, then 2400 and 2000 baud terminals, then TTYs, then empty ports. The MUX71 macro port count parameter can be set to exclude the empty ports and increase driver efficiency. This saving is especially important when a driver is to support both a 6671 and a 6676.

A subtable for the 6673/6674 multiplexer might be defined as follows.

```
MUX4      MUX73      (0,3),(01,02)
```

The multiplexer is defined with terminals attached to ports 0 and 3, and also with graphics consoles 1 and 2 defined on port 0.

#### MULTIPLEXER DEFINITION ENTRIES

INTERCOM recognizes four types of multiplexers, the 6671, 6676, 6673 and 6674, and two types of communications subsystems, the 791 (LCC) and 2550 (NPU). They are defined with the following macros.

```
MUX71      Number of ports
MUX76      Number of ports, baud rate
MUX73      (p0,p1),(gc01, . . . ,gc16)
MUX74      (p0, . . . p3),(gc01, . . . ,gc36)
MUXLCC     Number of ports
MUX2550    Number of ports
```

The parameter, number of ports, indicates the highest number port+1 which INTERCOM is to service on that multiplexer for a 6671, 6676, 2550 and 791. The parameters, pi, are the port numbers which INTERCOM is to service on that multiplexer for the 6673 and 6674. The parameters, gci, are the 274 graphics console numbers which INTERCOM is to service on that multiplexer for the 6673 and 6674. Only one multiplexer may be defined with graphics consoles, and up to six graphics consoles may be specified for each port on that multiplexer. In the two-digit graphics console number, the first digit indicates the port number, and the second digit the graphics console number on that port.

A 6671 with ports 0, 1, and 3 attached to data sets should be defined as

```
MUX71  4
```

A 791 with ports 0, 1, and 3 attached to data sets should be defined as

```
MUXLCC 4
```

A 2550 with ports 1, 2, and 5 attached to modems should be defined as

```
MUX2550 6
```

The MUX73, MUX74, and MUX76 macros generate all multiplexer subtable entries necessary to completely define those multiplexers. For the 6671, 2550 and 791, port definition entries should follow to define each individual port on the multiplexer. An example of a MUX table entry follows.

```
MUX76  24,300  (24 ports at 300 baud)
```

Baud rate parameter can be 110, 150, or 300. 300 baud is the maximum rate for any TTY on a 6676 multiplexer.

#### PORT DEFINITION ENTRIES

Currently, five types of ports are recognized by the CMR macros for a 6671 multiplexer. They are defined with the following macros.

```
TM3      Mode 3 terminal. Teletype model 33, 35, or 38, or CDC 713.
TMB4A    BCD mode 4A terminal. CDC 214, 217, 73X-12, or 200 User Terminal.
TMA4A    ASCII mode 4A terminal. CDC 214, 217, 73X-12, 200 User Terminal, or 711 Model A/B.
TM4C     Mode 4C terminal. CDC 711 Model C/D, or 714.
EMPTY    Empty port (not serviced by INTERCOM).
```

Seven types of ports are recognized by the CMR macros for a 791. They are defined with the following macros.

```
TM3      Mode 3 terminal. Teletype model 33, 35, or 38, or CDC 713.
TMB4A    BCD mode 4A terminal. CDC 214, 217, 73X-12, or 200 User Terminal.
TMA4A    ASCII mode 4A terminal. CDC 214, 217, 73X-12, 200 User Terminal, or 711 Model A/B.
TM4C     Mode 4C terminal. CDC 711 Model C/D, or 714.
```

TM4 Any mode 4 terminal. CDC 214, 217, 73X-12, 200 User Terminal, 711 Model C/D, or 714.  
 TM2 Mode 2 terminal. CDC 731-10, 732-10, or 733-10.  
 EMPTY Empty port (not serviced by INTERCOM).

Six types of ports are recognized by the CMR macros for a 2550. They are defined with the following macros.

TM3 Mode 3 terminal. Teletype model 33, 35, or 38, or CDC 713. The mode 3 automatic speed recognition option may be used by the 2550 only.  
 TMB4A BCD mode 4A terminal. CDC 214, 217, 73X-12, or 200 User Terminal.  
 TMA4A ASCII mode 4A terminal. CDC 214, 217, 73X-12, or 200 User Terminal.  
 TM4C Mode 4C terminal. CDC 711 Model C/D, or 714.  
 TM4 Any mode 4 terminal. CDC 214, 217, 73X-12, 200 User Terminal, 711 Model C/D, or 714.  
 EMPTY Empty port (not serviced by INTERCOM). 2550 port 0 must be empty.

The TM4 macro allows any mode 4 terminal to dial into a 791 port, and INTERCOM automatically determines the specific terminal type from the site address.

The port definition macros immediately follow the macro for the corresponding multiplexer. Each macro defines one port, beginning with port 0 as the first entry, the second is port 1, and so on. All ports through the highest to be serviced by INTERCOM on that multiplexer must be defined. Thus, if the number of ports parameter on the MUX71 macro is 10B, then 8 port definition macros must follow even though some may not be used. Unused ports should be defined with the EMPTY macro.

Port definition macros without parameters indicate that the ports are servicing normal dial-up telephone circuits. Site addresses for dial-up mode 4 terminals are assumed to be 1 for the TMA4A macro, 0 for the TMB4A macro, and 2 for the TM4C macro. It is possible also to define hard-wired connections by adding parameters to any mode 4 terminal macro (except TM4) or to a mode 3 port on a 2550 as described in the following text.

#### LINE SPEEDS AND HARD-WIRED TERMINALS

Any of the terminal macros may be used to specify different line speeds by adding a parameter to the macro. Omission of the parameter indicates 110 bps for mode 3 terminals, 2000 bps dial-up for mode 4 terminals, and 50000 bps for mode 2 terminals. Addition of the HW parameter causes INTERCOM to assume that the terminal is hard-wired. Mode 2 and wideband terminals are always hard-wired. Mode 3 terminals on a 2550 may be either hard-wired or dial-up. All other mode 3 terminals are dial-up. It is possible to define 4800 bps dial-up on 2550 ports.

TMxxx line-speed

where line-speed may be the following for each terminal type

TM3 110, 150, 300 (also 600 and 1200 for the 2550)  
 TM3 \* (The asterisk defines automatic-speed-recognition of a 110, 150, or 300 bps terminal. This option is used only by the 2550.)  
 TMx4x 2400, 4800, 9600.  
 TM2 2400, 4800, 9600, 50000.

#### MULTIDROP LINES

Any of the mode 4 terminal macros may be used to define multidrop (party-line) configurations by adding parameters to the macro call. The general form is the following.

TMx4x line-speed,(sa0,sa1,...,sal1),HW

A list of site addresses indicates the port is to service a multidrop line to which terminal at those site addresses may be connected. Up to 12 site addresses, 0 to 13B, may be specified in any order. Omission of the site address list causes INTERCOM to assume site address 0. The site address list will not be processed if the line speed parameter is omitted. For example, a BCD Mode 4A port with a 2400 bps party-line with six possible site addresses might be defined as follows.

TMB4A 2400,(5,1,0,10,11,6),HW



**MULTISTATION TERMINALS**

The mode 4C terminal macro, and the general mode 4 macro, may be used to define multistation terminals by adding parameters to the macro call. The general form is the following.

TMx4x            line-speed,(site-address-list),(sa1,sa2, . . . sa11),HW

A list of station addresses indicates the terminal has several CRT stations to be serviced. Printer stations must not be specified in the macro call. Up to 9 station addresses, 1-3, 5-7, 11-13B, may be specified in any order. Station addresses 4, 10B, 14B, are reserved for printer stations. Omission of the station address list causes INTERCOM to assume station address 1. The station address list is not processed for mode 4A terminals.

For example, a mode 4C port with three site addresses on a 4800 bps line, each of which may have four station addresses might be specified as follows.

TM4C            4800,(0,5,2),(6,1,2,9),HW

In this case, each site is assumed to have the identical station configuration.

**PORT DISTRIBUTIONS FOR LOW SPEED MULTIPLEXERS**

The chart indicates the hardware limitations for each low-speed multiplexer configuration. These limitations show the maximum number of mode 4 ports that can be defined and the maximum number of mode 3 ports that can be defined. Each mode 4 port can be a party-line port and support more than one terminal.

For each hardware configuration, the chart indicates the best estimates of what the software will support satisfactorily. Under heavy loads, terminals may suffer some degradation.

Multiplexer Configuration	Hardware Limits		Software Limits			
			2000-2400 bps	4800 bps	9600 bps	
One PPU	Mode 4	Mode 3	Mode 4	Mode 4	Mode 4	Mode 3
6671	16		16	16	2-4	
		16				16
2X6671	32		28-32	Not Supported	Not Supported	
	16	16	14-16	Not Supported	Not Supported	16
		32				32
6676		64				64
2X6676		128				100-128
6671+6676	16	64	16	10-12	Not Supported	64 †
		80				80

† The number of 110 baud lines that can be supported on the 6676 is a maximum of 25 to 40 if any 4800 baud lines are defined for the 6671. The greater the number of ports used on the 6671, the fewer that can be used on the 6676. In fact, if all 16 ports are used on the 6671 and any are run at 4800 baud, the 6676 will not be serviced. If either 2400 or 4800 baud mode 4 terminals are on the 6671 and 300 baud mode 3 terminals are on the 6676, fewer than 25 ports are supported on the 6676.

## DRIVER TYPE SELECTION

Each hardware channel to be serviced by INTERCOM low speed is processed by a single dedicated PPU. The INTERCOM low-speed driver assembles differently depending on the channel multiplexer and terminal configuration. Depending on the system, it may be necessary to have more than one variant of the driver assembled in the system since INTERCOM low-speed may be supporting more than one channel.

A \*DEFINE directive is required for each type of driver to be assembled.

The \*DEFINE directive, the configuration supported by the variant, and the PPU program name of the variant are indicated in the following table where C represents a CRT, T a TTY; 71 is a 6671 and 76 is a 6676. S represents a special stimulator communication variant.

<u>Directive</u>	<u>Configuration</u>	<u>PPU Name</u>
*DEFINE,CT71	One or two 6671s; CRT terminals with speeds up to 2400 bps (BCD and/or ASCII) or One 6671; CRT terminals with speeds up to 4800 bps (BCD and/or ASCII). TTYs as well as CRT terminals may be on 6671 ports.	1Z2/8Z2
*DEFINE,CT7176	One 6671 and one 6676. 6671 has CRT terminals with speeds up to 4800 bps and may also have TTY ports. 6676 has TTY ports.	1Z3/8Z3
*DEFINE,T76	One or two 6676s; each multiplexer has TTYs only.	1Z6/8Z6
*DEFINE,CT96	One 6671; lowest four ports of multiplexer are supported and may have TTYs or CRT terminals with speeds up to 9600 bps.	1Z7/8Z7
*DEFINE,ST76	One simulated 6676, TTYs only; variant only communicates with software simulation programs SIP/VSM/1VG.	1Z8/8Z8

With one exception, given the \*DEFINE directive, the driver will be assembled for the maximum number of multiplexers that can be supported by that variant. Initialization code within the driver will set internal tables to reflect the actual configuration. For example, the T76 driver is assembled to support two 6676 multiplexers. If this variant is called to support a channel with only one 6676, it will initialize properly to handle this situation. The exception is 1Z3/8Z3, the variant which drives both a 6671 and a 6676 on the same channel. If either mux is down, or logically OFF in the EST, and out of service, a driver variant for the single multiplexer is required. Sites with a configuration which includes a channel shared by a 6671 and a 6676 should install three driver variants in order to drive either or both multiplexers. These variants include the following.

1Z3/8Z3	*DEFINE,CT7176
1Z2/8Z2	*DEFINE,CT71
1Z6/8Z6	*DEFINE,T76

Two CRT terminal drivers (1Z2 and 1Z3) will initialize to process terminals at either a 2400 bps rate or a 4800 bps rate as is necessary.

Note that 1M1 cannot support two 6671s on a single channel if any of the ports must be driven at 4800 bps.

Parameters IP.CTCT, IP.CTCT4, and IP.CTCT9 indicate the maximum number of ports active. The parameter for the fastest line speed is used in determining the maximum for each driver variant. This number is doubled if the driver is running at 2XPP speed, which requires a restriction that two times this maximum is less than 73B (77B-SLOTRAN) if the driver is to be run at 2XPP speed with CRTs.

The \*DEFINE,CT71 directive for the low-speed driver is located near the end of the PL12I installation deck.

For each additional variant of the low-speed driver required, the following sets of cards must be inserted in PL12I at the positions designated by comments contained in that deck (refer to Installation Procedures).

Let x represent the driver type, such as T76 or T7176 (refer to Driver Type Selection).

In the control statement record, insert

```
UPDATE(P=NEWPL,C=x,Q) UPDATE DRIVER x  
COMPASS(I=x,S=IPTEXT,S=PPTEXT,S=SDDTEXT,L=0) ASSEMBLE DRIVER x
```

As an input record to the preceding, insert the following directives.

```
*IDENT x
*DEFINE x
*COMPILE 1M1
=CWEOR,0
```

The UPDATE deck name for the low-speed driver is 1M1; however, each variant of the driver produced will have a different name of the form 1Zx where x is a number indicating the terminal mix supported by that driver as indicated in the table under driver type selection. Similarly, the driver's primary overlay, 8Zx, will have a name indicating the terminal mix supported. An additional overlay, 9Z1, is present in all driver variants.

When INTERCOM is first initiated, the INTERCOM initialization routine, 111, initiates the drivers as dictated by the multiplexers defined in the EST and the port definitions defined in the multiplexer subtables. If all equipments (multiplexers) on a channel are turned off when INTERCOM is initiated, no driver is initiated to service that channel; however, the multiplexer subtables for all of the equipment will be examined and initialized by 111.

The user should make certain that only one EST entry points to each multiplexer subtable whether the equipment is on or off.

Installation deck PL121 also will compile the relocatable multiuser job subroutines (deckname MUJSUBS). Deck PL12E will not add them to the running system for reasons of size and expected infrequency of use. MUJSUBS always must be included on the COMPILE file, however, when EDITOR is compiled and loaded, so that references to the muj subroutines from EDITOR are satisfied. If a full UPDATE is done, the subroutines are included on the COMPILE file. If an UPDATE,Q is done and the EDITOR is to be modified, the UPDATE input must include a \*COMPILE MUJSUBS. (EDITOR does not use FTNMUJ or COBOMUJ, the decknames for the FORTRAN Extended and COBOL muj preprocessors.)

After the password files are established and the time has been initialized, INTERCOM should be brought up at control point zero with the console type-in INTERCOM. The INTERCOM system is then ready to service remote terminal users.

#### COMMAND TABLE STRUCTURE (1CI OVERLAY 2CS — COMMON DECK COMTBL)

Prior to INTERCOM installation, release values in the command table in 2CS may be changed or a new command or multi-user-job entry may be added. The command table is split into four parts based on the length of the command name. New entries should be inserted at the following locations.

1- or 2-character name	*I,COMTBL.12
3- or 4-character name	*I,COMTBL.35
5- or 6-character name	*I,COMTBL.58
7-character name	*I,IN40844C.14

The four command types each have an entry-definition macro as follows.

COM2CC	Defines a command processed by 2CC.
MUJ	Defines a multiuser job.
COM1LX	Defines a remote-batch command processed by 1LX.
REMOTE	Defines a command which manipulates queue files or executing jobs.

A command-definition entry has the general form

name MACRO parameters

where name is the command name, such as, ON, and MACRO is one of the above macro names.

#### COM2CC MACRO

The COM2CC macro defines a command which is processed by an independent routine in overlay 2CC. The format is as follows.

name	COM2CC L=1, P=p, B=b, MP=mp, ADDR=ad
1	YES User must be logged in to use this command NO User need not LOGIN if at a hardwired terminal Default YES
p	YES Command may be used while in a pause state NO Command may not be used while in a pause state Default NO

b	YES	Command allowed only at a batch terminal
	NO	Command allowed from any terminal type
	Default	NO
mp		Maximum number of parameters which may follow command verb; Range 0-5. If MP is specified, even MP=0, parameters in the input line are counted. If the number of parameters exceeds mp, the line is rejected as a format error. Do not specify MP when commands contain parameters over 7 characters or for commands such as MESSAGE for which parameters are meaningless.
ad		2CC address (routine name) where this command is processed. If the AD parameter is omitted, a routine with the same name as that of the command is assumed.

#### MUJ MACRO

The MUJ macro defines a multiuser job. A corresponding entry must be made in muj table of 1QP. The format is as follows.

name	MUJ ORD1QP=ord
ord	1QP MUJ ordinal. EDITOR=1, HELLO7=2; others should proceed sequentially from 3

#### COM1LX MACRO

The COM1LX macro defines a command processed by 1LX, and controls parameter processing for the command. The format is as follows.

name	COM1LX B=b, P=p, MP=mp, ORD1LX=ord, PRE=pre, POST=post, MU=mu
b	Same as for COM2CC, except default =YES
p	Same as for COM2CC
mp	Same as for COM2CC
ord	1LX command ordinal. An entry must be added at installation to the 1LX jump table for each new COM1LX command.
pre	Address (name) of 2CC subroutine which does preprocessing (prior to extraction and validation of equipment mnemonic) for this command.
post	Address (name) of 2CC subroutine which does postprocessing (after equipment validation but before passing directive to 1LX) for this command.
mu	Minimum unit mnemonic for which command is valid. ALL < CR < CP < LP.

#### COM1LX NOTES

Either PRE or POST may be specified, but not both. If neither is specified, only the directive ordinal and the equipment number is passed to 1LX. Refer to the IMS for further information.

#### REMOTE MACRO

The REMOTE macro defines commands which manipulate the user's queue files and execution jobs, specifically the commands DROP, KILL, DIVERT, EVICT, and PRIOR. Adding such an entry requires modifications to the 2CC routine REMOTE. Anyone contemplating this course should consult the IMS.

#### MUJ TABLE STRUCTURE (1QP)

Each multiuser job as defined in the command table of 2CS also must be defined in the muj table of 1QP, MUJTABL. The position of an entry in MUJTABL is defined as the 1QP muj ordinal. Entries are made with the macro MUJTBL, at \*B 1QP.599.

MUJTBL	name,fl,swpin,swpout,editor
name	Name of the muj
fl	Field length of muj (actual value)

swpin	Delay, in 1 CI cycles (depends on IP.TICI, released for 1/2 second), between discovery of need to swap in the muj and actual entry into the scheduling queue. This value increases response time to muj requests (when the muj is swapped out) but allows requests to accumulate, so that when the muj is in, it is more likely to process multiple users. Maximum of 4095.
swpout	Delay, in 1 CI cycles, between discovery of need to swap out muj and actual swap out. A high value setting essentially dedicates the muj at a control point.
editor	1 muj EDITOR 0 otherwise

The parameters swpin, swpout, and editor may be null, and default values 1, 0, and 0, respectively, are assumed.

#### TBL ASSEMBLY OPTIONS

Ten TBL command ordinals (14-23) are reserved for users to add routines to TBL. To add a routine with entry point xxx and command ordinal 14, change the fourteenth entry of TBL table TABLE to

CON    xxx

The TBL command ordinal is an index into ICPLIB. TBL tests bits 0 and 1 of table ICPLIB (12-bit entries) to determine if checks should be made for the calling program. If bit 0 is set, the calling program is a system library program. If bit 1 is set, the calling program is at an INTERCOM control point.

#### TABLE CHANGES AND RELEASE SETTINGS

Changes to the tables in routines 2CS, 1QP, and TBL should be included in the UPDATE record at the directive \*/ADD CORRECTIONS HERE in installation deck PL12I. The following list shows the release values and UPDATE identifiers.

* MUJTABL FOLLOWS. ALL DEFINED MUJ#S ARE INCLUDED. THEY MUST			1QP	594
* PHYSICALLY OCCUR IN THE SAME ORDER AS THEY DO IN 1SJ#S TABLE.			1QP	595
* MUJTABL	BSS	0	1QP	596
	MUJTBL	EDITOR,40000,0,2,1	1QP	597
			1QP	598
VERB12	EQU	*	COMTBL	2
C	COM1LX	P=NO,MP=0,ORD1LX=17B	COMTBL	3
E	COM1LX	P=NO,MP=1,ORD1LX=14B,PRE=END	COMTBL	4
G	COM1LX	P=YES,B=NO,MP=1,ORD1LX=3,PRE=GO	COMTBL	5
GO	COM1LX	P=YES,B=NO,MP=1,ORD1LX=3,PRE=GO	COMTBL	6
H	COM2CC	L=NO,P=NO,B=YES,MP=2	COMTBL	7
M	COM2CC	L=NO,P=YES,B=NO,ADDR=MESSAGE	IN40844K	7
ON	COM1LX	P=NO,MP=1,ORD1LX=1	COMTBL	9
			COMTBL	10
VERB34	EQU	*	COMTBL	13
BSP	COM1LX	P=NO,MP=2,ORD1LX=10B,MU=CP,POST=BSP	COMTBL	14
CRT	COM2CC	L=NO,P=NO,B=YES,MP=1	COMTBL	15
DMP	COM2CC	L=YES,P=NO,B=NO,MP=2	IN41257	4
DROP	REMOTE	P=YES,MP=1,REMORD=0,SUBORD=1	COMTBL	18
EFL	COM2CC	L=YES,P=NO,B=NO,MP=1	COMTBL	19
END	COM1LX	P=NO,MP=1,ORD1LX=14B,PRE=END	COMTBL	20
ETL	COM2CC	L=YES,P=NO,B=NO,MP=1	COMTBL	21
KILL	REMOTE	P=NO,MP=1,REMORD=0,SUBORD=0	COMTBL	22
LOCK	COM2CC	L=NO,P=YES,B=NO,MP=1	COMTBL	23
OFF	COM1LX	P=NO,MP=1,ORD1LX=2	COMTBL	24
READ	COM2CC	L=NO,P=NO,B=YES,MP=2	COMTBL	26
REP	COM1LX	P=NO,MP=2,ORD1LX=13B,MU=CP,POST=REPEAT	COMTBL	27
REW	COM1LX	P=NO,MP=1,ORD1LX=11B,MU=CP	COMTBL	28
RFL	COM2CC	L=YES,P=NO,B=NO,MP=1,ADDR=EFL	COMTBL	29
RTN	COM1LX	P=NO,MP=2,ORD1LX=12B,MU=CP,POST=RTN	COMTBL	30
SUP	COM1LX	P=NO,MP=1,ORD1LX=15B,MU=LP	COMTBL	31
TAPE	COM2CC	L=YES,P=YES,B=NO,MP=1	COMTBL	32
WAIT	COM1LX	P=NO,MP=1,ORD1LX=4	IN40844C	13
			COMTBL	35
			COMTBL	36
VERB56	EQU	*	COMTBL	37
			COMTBL	38
COMTBL 39				
CONTIN	COM1LX	P=NO,MP=0,ORD1LX=17B	IN40844B	5
DEFINE	COM1LX	P=NO,MP=5,ORD1LX=5,MU=CP,PRE=DEFINE	COMTBL	41
DIVERT	REMOTE	P=NO,MP=4,REMORD=3,SUBCRD=1,CP=PARM3,MO=ALL	COMTBL	42
EDITOR	MUJ	ORD1QP=1	COMTBL	43
EVICT	REMOTE	P=NO,MP=2,REMORD=1,SUBORD=0,QP=PARM2,MQ=ALL	COMTBL	44
HELLO7	MUJ	ORD1QP=2	S402580	3
			COMTBL	47
LGIMORD	EQU	* FWA OF LOGIN ENTRY	IN41353	5
LOGIN	COM2CC	DUMMY ENTRY - ONLY NAME USED	COMTBL	49
LGOTORD	EQU	FWA OF LOGOUT ENTRY	IN41353	6
LOGOUT	COM2CC	DUMMY ENTRY - ONLY NAME USED	IN41353	7
			COMTBL	50
PRIOR	REMOTE	P=NO,MP=3,REMORD=2,SUPORD=0,QP=PARM3,MO=OUTPUT	COMTBL	51
REDUCE	COM2CC	L=YES,F=NO,B=NC,MP=1	COMTBL	53
REVERT	COM2CC	L=NO,P=NO,B=YES,MP=1	COMTBL	54
SAVEFL	COM2CC	L=YES,P=NO,B=NO,PF=1	COMTBL	56
SWITCH	COM2CC	L=YES,P=YES,B=NO,MP=1	COMTBL	58
			COMTBL	59
VERB7	EQU	*	COMTBL	60
			COMTBL	61
COMMENT	COM2CC	L=NO,P=YES,B=NO	COMTBL	63
MESSAGE	COM2CC	L=NO,P=YES,B=NO	IN40844C	14
			COMTBL	66
TBLEND	EQU	* LWA+1 OF TABLE = LWA OF VERB7 SECTION	COMTBL	67

EDITOR INSTALLATION PARAMETERS

EDITOR uses two common decks, IPFTN (FORTRAN) and IPCOM (COMPASS), to contain installation parameters. Generally, a change to one common deck requires a corresponding change to the other. With the exception of arrays which must be dimensioned for FORTRAN in common deck IPFTN, the values of installation parameters are not defined in IPFTN. IPFTN merely allocates storage for these definitions. The definitions are DATA statements in the BLOCK DATA subprogram IPFILL.

IPCOM contains EQUs which define the installation parameters. Since many parameters are of such a nature that a change in one implies a change of another, a dependency chart is included in this subsection to aid the installation.

A summary of steps to take to change an EDITOR installation parameter follows.

1. Change the DATA statement in IPFILL or the EQU in IPCOM, or both, as indicated by the parameter description.
2. Consult the dependency chart for any dependent installation parameters that require change, and change them as in step 1.
3. Consult the dependency chart for dimensions of arrays in IPFTN. If they are affected, change them as indicated in the table, Array Dimensions in IPFTN.

Additionally, EDITOR has the following installation parameter defined in common deck INTCOM.

IP.FTNTS	CEQU	0	Specifies the FORTRAN Extended 4 installation default compiler as OPT=0
		1	Specifies the FORTRAN Extended 4 installation default compiler as time-sharing

Any changes which cause the size of the EDITOR to increase may require an increase in the field length defined for EDITOR in the MUJTABL for 1QP. The following list shows the release values and UPDATE identifiers for IPFILL, IPCOM, and IPFTN.

IPCOM

* THE FOLLOWING SYMBOLS MUST BE DEFINED FIRST, SINCE THEY ARE USED TO				IPCOM	3
* DEFINE OTHER SYMBOLS BELOW				IPCOM	4
NPRUS	EQU	2	NUMBER OF PRUS IN EDIT FILE *WINDOW*	I410095	3
NTBSMAX	EQU	10	MAX. NUM. OF TAB SETTINGS ALLOWED	I410095	4
NSINDEX	EQU	20	SIZE OF EDITFIL INDEX	I410079	1
* INSTALLATION PARAMETERS				IPCOM	86
* SEE ALSO- ADDITIONAL PARAMETERS DEFINED AT VERY BEGINNING OF IPCOM				IPCOM	87
NBBS	EQU	2	NUMBER OF BIG BUFFERS	IPCOM	88
NSBB	EQU	257	SIZE OF BIG BUFFER (WITHOUT FET)	I410095	5
NPBS	EQU	3	NUMBER OF POOL BUFFERS	IPCOM	90
NUSERS	EQU	30	MAXIMUM NUMBER OF ATTACHED USERS	IPCOM	91
NEDFETS	EQU	10	MAXIMUM NO. OF EDIT FILE FETS	IN40815	42
NUAS	EQU	3	NUMBER OF USER AREAS	IPCOM	93
NSUA	EQU	JOPRU+NPRUS*64		IN40815	43
* EXCLUDES THE TAB POSITIONS, THE INDEX, AND THE RJLNKS AREA				IPCOM	95
NSRJLNK	EQU	10	SIZE OF RU LINKAGE AREA	IPCOM	97
NUASIZE	EQU	JRULNKS+NSRJLNK		IN40815	44
*				IPCOM	98
* DEBUG OPTION - IF DEBUG EQU 1, DEBUGGING CODE IS ASSEMBLED				IPCOM	99
NDEBUG	EQU	0	NO DEBUGGING CODE	IPCOM	100

IPFILL

C	IMPORTANT NOTE - THE VALUE OF THE FOLLOWING VARIABLES IS DEPENDENT	IT30062	31
C	ON THE VALUE OF THE INSTALLATION PARAMETER NPRUS -	IT30062	32
C	NSUA,	IT30062	33
C	JTABS, UNDXHDR, JINDEX, JRJLNKS	IT30062	34
C	IF NPRUS IS NOT EQUAL TO ONE, THE FOLLOWING MODIFICATIONS MUST BE	IT30062	35
C	MADE. FOR EACH OF THE VARIABLES LISTED ABOVE, ADD	IT30062	36
C	THE VALUE (NPRUS-1)*64.	IT30062	37
	DATA JTEXT1/0/,JTEXT2/2/,JFLAGS1/4/,JFLAGS2/5/,JPBW/6/,JLNBUF/17/,	IN40815	741
	X JCPRU/69/,UTABS/197/,JNDXHDR/199/,JINDEX/200/,JRJLNKS/220/	IN40815	742
	DATA NLINE/6L000100/,NINCR/10/,NUAS/3/	EDITOR	4251
	DATA NBBS/2/,NPBS/3/,NUSERS/30/,NEDFETS/10/,	IN40815	746
	X NSUA/197/,NSINDEX/20/,NSRJLNK/10/,NTBSMAX/10/,XNPCENT/0.9/,	IN40815	747
	X NUASIZE/230/,NPRUBUF/128/	IN40815	748
C	SEE NOTE ABOVE IF NPRUS IS CHANGED.	IT30062	38
	DATA NPRUS /2/	I410095	8
	DATA NTABFTN/1L;/,NTBSFTN/5/,NCHFTN/72/,	F7840ED	9
	X FTNTABS/00070012001500200023B/	F7840ED	10
	DATA NTABCOM/1L;/,NTBSCOM/3/,NCHCCM/72/,	EDITOR	4256
	X COMTABS/00130022004400000000B/	EDITOR	4257
	DATA NTABCOB/1L;/,NTBSCOB/5/,NCHCCB/72/,	EDITOR	4258
	X CCBTABS/00100014002000240030B/	EDITOR	4259
	DATA NTABALG/1LS/,NTBSALG/5/,NCHALG/72/,	IT30125	1
	X ALGTABS/00070012001500200023B/	EDITOR	4261
	DATA NCHBAS/150/	IN41370	113
	DATA NTABDEF/1L;/,NTBSDEF/5/,NCHDEF/72/,	F7840ED	11
	X DEFTABS/00070012001500200023B/	F7840ED	12

IPFTN

C	THIS COMMON DECK CONTAINS ALL INFORMATION CONCERNING THE FORMAT OF	IPFTN	3
C	THE EDITOR COMMON AREA (ECA) AND INSTALLATION PARAMETERS	IPFTN	4
C	THIS STATEMENT DEFINES THE EDITOR COMMON AREA	IPFTN	8
C	ALL OF THE CELLS IN COMMON BLOCK /ECA/ ARE FILLED BY (ECAFILL)	IPFTN	9
	COMMON /ECA/ MLRROR, MEM(1), MRA,	IN40815	5
	X MPTR1, MPTR2, MCNT,	IPFTN	11
	1 MIC1, MCASE, MSTATE, MCASE2, MSTATE2, MUNMBR, MOASUB, MEFPTR, MACTN,		
	2 MCMLPT, MTEBCNT, MTEB(S2), MBBMA(2), MPBMA(3), MBBS(526), MPBS(384),	I410095	1
	3 MEDFST(80), MMUJTBL(159), MPBWD, MBBWD, MUAS(690)	IN40815	7
	X, IEDFET, ITPRUS, IWDSWRT, IUFRUS	IN40815	8
C	NUMBERS OF WORDS IN USER AREA (RELATIVE TO ZERO)	IPFTN	37
	COMMON/ IPFTN/ JTEXT1, JTEXT2, JFLAGS1, JFLAGS2, JPBW, JLNBUF, JCPRU,	IPFTN	38
	X JTABS, JNDXHDR, JINDEX, JRJLNKS	IPFTN	39
C	NUM. OF WDS IN EDIT FILE WINDOW- INITIALIZED AT BEGINNING OF EDITOR	IPFTN	62
	COMMON /IPFTN/ NPRUBUF	IPFTN	63
C	INSTALLATION PARAMETERS	IPFTN	64
	COMMON/ IPFTN/ NLINE, NINOR, NBBS, NP8S, NUAS, NUSERS,	IN40066	1
	X NSUA, NSINDEX, NSRJLNK, NTBSMAX, XNPCENT, NPRUS, NUASIZE, NEDFETS	IN40815	12
C	(FORTRAN) FORMAT	IPFTN	67
	COMMON/ IPFTN/ NTABFTN, NTBSFTN, NCHFTN, FTNTABS(1)	IPFTN	68
C	(COMPASS) FORMAT	IPFTN	69
	COMMON/ IPFTN/ NTABCCM, NTBSCOM, NCHCOM, COMTABS(1)	IPFTN	70
C	(COBOL) FORMAT	IPFTN	71
	COMMON/ IPFTN/ NTABCOB, NTBSCOB, NCHCOB, COBTABS(1)	IPFTN	72
C	(ALGOL) FORMAT	IPFTN	73
	COMMON/ IPFTN/ NTABALG, NTBSALG, NCHALG, ALGTABS(1)	IPFTN	74
C	(BASIC) FORMAT	IPFTN	75
	COMMON/ IPFTN/ NCHBAS	IPFTN	76
C	DEFAULT FORMAT	IPFTN	77
	COMMON/ IPFTN/ NTABDEF, NTBSDEF, NCHDEF, DEFTABS(1)	IPFTN	78

In table II-12-1, -\* in the Range column indicates where a parameter has essentially no absolute upper limit. The installation determines the practical upper limit based on considerations such as EDITOR size and expected number of users.



TABLE II-12-1. EDITOR INSTALLATION PARAMETERS

Parameter Name	Defined In		Description	Range	Release Value
	IPFILL	IPCOM			
NLINE	X		Default first line number for CREATE, EDIT, RESEQ	6L000001 to 6L999999	6L000100
NINCR	X		Default line number increment for ADD, CREATE, EDIT, RESEQ	1-999998	10
NUAS	X	X	Number of user area buffers	1-* Large number decreases response time if there are many users	3
NBBS	X	X	Number of big buffers (used for EDIT, SAVE, RUN)	1-* Increase if many EDITS, SAVES, RUNS anticipated	2
NPBS	X	X	Number of pool buffers. Each is 64*NPRUS words	2-* Increase when heavy file modifications or long text lines expected, generally NPBS ≥ NUAS	3
NUSERS	X	X	Maximum number of users simultaneously using EDITOR	1-* Vary with expected usage of EDITOR	30
NPRUS	X	X	Number of 64-word PRUs in one block in edit file  If NPRUS is larger than 12D, EDITOR will not use the CMU hardware, regardless of the setting of NOS/BE 1 installation parameter IP.CMU.	1-* Large number decreases response time for commands which process large files, but it also increases amount of central memory required for EDITOR by 64 words for each pool buffer and 64 words for each user area buffer	2
NSUA	X	X	Size of user area; must be modified in IPFILL if NPRUS is changed. NSUA=69+ 64*NPRUS. Size does not include portion of user area used for tabs, return jump links, and edit file index	133-*	197
NUASIZE	X	X	SIZE OF USER AREA including areas for tabs, return jump links and edit file index.	133-*	230
NPRUBUF	X		Number of words in one edit file block. Must be 64*NPRUS	64-*	128
JTABS	X		Number of word in user area which holds tab values; must be modified in IPFILL if NPRUS is changed. JTABS= 69+64*NPRUS	131-*	197

TABLE II-12-1. EDITOR INSTALLATION PARAMETERS (Contd)

Parameter Name	Defined In		Description	Range	Release Value
	IPFILL	IPCOM			
JNDXHDR	X		Number of index header word in user area; must be modified in IPFILL if NPRUS is changed. JNDXHDR= JTABS+(NTBSMAX+4)/5	132-*	199
JINDEX	X		Number of first word in edit file index in user area; must be modified in IPFILL if NPRUS is changed. JINDEX= JNDXHDR+1	133-*	200
JRNLNKS	X		Number of first word in return jump link area in user area; must be modified if NPRUS is changed. JRNLNKS= JINDEX+NSINDEX	153-*	220
NSINDEX	X	X	Number of index entries for each user's edit file	1-* Increase for editing very large files	20
NTBSMAX	X	X	Maximum number of tab settings permitted by FORMAT command	1-509 Must be > NTBSFTN, NTBSCOM, NTBSCOB, NTBSALG, NTBSDEF	10
XNPCENT	X		Percent to which each block of user's edit file is filled by EDIT (Padding factor)	.01-1.00 Decrease if heavy file modification is expected	.90
NTABFTN	X		FORTRAN tab character	1 LA-1 L;	1L;
NTABCOM	X		COMPASS tab character	1 LA-1 L;	1L;
NTABCOB	X		COBOL tab character	1 LA-1 L;	1L;
NTABALG	X		ALGOL tab character	1 LA-1 L;	1L\$
NTABDEF	X		Default tab character	1 LA-1 L;	1L;
NTBSFTN	X		Number of FORTRAN tabs defined	0-509	5
NEDFETS	X	X	Number of FETs used to attach a user's editfile.	1-*	10
NTBSCOM	X		Number of COMPASS tabs defined	0-509	3
NTBSCOB	X		Number of COBOL tabs defined	0-509	5
NTBSALG	X		Number of ALGOL tabs defined	0-509	5
NTBSDEF	X		Number of Default tabs defined	0-509	5
NCHFTN	X		Maximum number of characters in FORTRAN line	1-510	72
NCHCOM	X		Maximum number of characters in COMPASS line	1-510	72

TABLE II-12-1. EDITOR INSTALLATION PARAMETERS (Contd)

Parameter Name	Defined In		Description	Range	Release Value
	IPFILL	IPCOM			
NCHCOB	x		Maximum number of characters in COBOL line	1-510	72
NCHALG	X		Maximum number of characters in ALGOL line	1-510	72
NCHDEF	X		Maximum number of characters in default format	1-510	72
NCHBAS	X		Maximum number of characters in BASIC line		150
FTNTABS	X		Consecutive stream of bits, each 12 define a tab position for FORTRAN format. Must be ascending order	1-511 (each tab)	00070012001500200023B
COMTABS	X		Same as above, for COMPASS	1-511	00130022004400000000B
COBTABS	X		Same as above, for COBOL	1-511	00100014002000240030B
ALGTABS	X		Same as above, for ALGOL	1-511	00070012001500200023B
DEFTABS	X		Same as above, for Default format	1-511	00070012001500200023B
NSBB		X	Size of big buffers used for EDIT, SAVE, RUN (does not include FET)	64-* Increase for very large files	157
NDEBUG		X	Flag controls debugging mode.  (Refer also to Multiuser Job Installation Parameter MDEBUG)	0 = off 1 = on	0

TABLE II-12-2. EDITOR Array Dimensions in IPFTN

Array Name	Usage	Array Dimension
FTNTABS	FORTTRAN tabs	(NTBSFTN+4)/5
COMTABS	COMPASS tabs	(NTBSCOM+4)/5
COBTABS	COBOL tabs	(NTBSCOB+4)/5
ALGTABS	ALGOL tabs	(NTBSALG+4)/5
DEFTABS	Default tabs	(NTBSDEF+4)/5
MMUJTBL	Storage needed by muj subroutine tables	4*NUSERS + NBBS + 10 + NUAS + 6* (NPBS+1)
MUAS	User area buffers May never exceed 4095 decimal	NUAS* (size of full user area) where: (size of full user area) (NTBSMAX+4)/5 + NSINDEX +1 + NSUA + NSRJLNK Note: NSRJLNK should not have to be changed by an installation
MBBS	Big buffers	NBBS*NSBB + NBBS*6
MPBS	Pool buffers	NPBS*64*NPRUS
MBBMA	Big buffer management area	NBBS
MPBMA	Pool buffer management area	NPBS

TABLE II-12-3. EDITOR DEPENDENCIES

If changed	Check parameters in IPFILL and/or IPCOM	Check arrays in IPFTN
NLINE NINCR NUAS NBBS NPBS NUSERS† NSINDEX NTBSMAX XNPCENT NTABFTN NTABCOM NTABCOB NTABALG NTABDEF NTBSFTN NTBSCOM NTBSCOB NTBSALG NTBSDEF NCHFTN NCHCOM NCHCOB NCHALG NCHDEF NCHBAS FTNTABS COMTABS COBTABS ALGTABS DEFTABS NSBB NDEBUG NPRUS	JRJLNKS JNDXHDR, JINDEX, JRJLNKS  NTBSMAX, FTNTABS NTBSMAX, COMTAABS NTBSMAX, COBTABS NTBSMAX, ALGTABS NTBSMAX, DEFTABS  NTBSFTN NTBSCOM NTBSCOB NTBSALG NTBSDEF  JTABS, JNDXHDR, JINDEX, JRJLNKS, NSUA	MMUJTBL, MUAS MBBS, MBBMA MPBS, MPBMA, MMUJTBL MMUJTBL MUAS MUAS  FTNTABS COMTABS COBTABS ALGTABS DEFTABS  FTNABS COMTABS COBTABS ALGTABS DEFTABS MBBS MDEBUG † † MPBS, MUAS
† When NUSERS is increased, the user should also consider changing the size of the TERMIN and TERMOUT tables in the muj subroutine MUJSUBS. Refer to the INTERCOM 4 Multiuser Job Capability Programming System Bulletin under the heading Changing Size of TERMIN and TERMOUT. † † Refer to Multiuser Job Installation Parameters.		

EDITOR DEBUG CODE

If EDITOR encounters hardware and/or software problems, a diagnostic printout is produced. If the problem is considered fatal, all EDITOR users are detached. The content of the diagnostic printout depends on the error encountered and the setting of NDEBUG. In any event, the diagnostic printout should accompany any PSR relating to a MUJ SYSTEM ERROR. Refer also to MDEBUG in the following subsection.

MULTIUSER JOB INSTALLATION PARAMETERS

The multiuser job (muj) subroutines use two common decks, MUJCOM and CMUJCOM. Both contain storage allocation for an array, ECSBUF. The MUJCOM deck in FORTRAN code contains a DIMENSION statement; the CMUJCOM deck in COMPASS code contains a BSS statement. This array is used by the muj peripheral processor routines, FAD, to read information from extended core storage (ECS). Array length must be (n\*64+1) central memory words. The value of n can be selected by the installations, depending on the expected use of ECS for storage of user swap files (if ECS is used, n should be at least 2) and on the number of local files allowed for an INTERCOM user. As a guide, n may be increased by one for each 20 local files allowed per user. The upper limit for n is dependent on the amount of storage used for the ECS buffer in the muj, and the size of the swap buffer in FAD.

The peripheral processor routine FAD contains two parameters relevant to allocation of space for ECSBUF. ECSBFLN (near FAD.659) is a COMPASS EQU instruction. It must be equated to the number of central memory words in the ECSBUF array. SWAPBF (near FAD.650) is a table FAD uses to read the ECSBUF array into PP memory. The value of ECSBFLN, and thus the size of the ECSBUF array in MUJCOM and CMUJCOM, must not be greater than 1 + (length of SWAPBF)/5.

Symbol MDEBUG in common deck CMUJCOM controls muj debugging code (0=off, 1=on). It should be set to 1 if the EDITOR installation parameter NDEBUG is set to 1.

In the routine MUJFILL, the two constants NACOUNT and THRSHLD control the accounting of muj time. The value of NACOUNT determines how frequently the accounting information for a muj is obtained from the system and distributed to users attached to the muj. NACOUNT is the maximum number of user switches performed on any given user before accounting is done. Accounting is always performed on user exit from the muj. NACOUNT must be set greater than or equal to 1 and defaults to 50 decimal. The value of THRSHLD determines the minimum number of CP seconds accumulated before accounting is posted to the user. As THRSHLD is set to smaller values, accounting is more accurate, overhead is increased, and THRSHLD has more meaning. THRSHLD defaults to 5 decimal.

## INSTALLATION PROCEDURES

Installation job decks PL12I and PL12E can be obtained from the Installation Decks program library, using the procedure outlined in part I, section 1 of this document.

Deck PL12I assembles the released program library adding the created binary to the PL tape as supplemental files. The release tape does not contain assembled binary. Deck PL12E uses EDITLIB to enter the binary created by deck INTCM1 into the running system. Deck PL12I requires modification if the single default low speed driver type is not the correct variant or multiple low-speed driver types are desired. If the hardware configuration does not include at least one 6671 or 6676 (for example, LCC only system), assembly of the low-speed multiplexer driver can be avoided by including an =DEFINE NO1M1 directive in the input record to the update of the installation deck PL to secure job PL12I.

Deck PL12O, applicable only in the user library method of installation, allows regeneration and replacement of absolute overlays in the user library plus creation of a new PL12 tape.

With deck PL12I, one variant of the PP multiplexer low-speed driver is produced which drives mode 4 terminals and Teletypes on the 6671 multiplexer.

Deck PL12E suggests CM residency for selected PP routines. Sites having ECS may wish to move some of these PP routines to ECS by employing the method discussed in the System ECS Resident Routine and Library portion of part II, section 1. Once PL12E has been run, job DST3 can be run to capture a deadstart tape containing INTERCOM. Decks PL12E and DST3 need not be run if the user library installation process is being followed.

However, for installation with a 6673 or 6674, overlays 2WB, 3WB, 4WB, and 5WB (WB-drive) should be CM resident; otherwise, the response time for the high-speed terminals (especially with interactive graphics consoles) is adversely affected.

No INTERCOM PP programs are required to be CM resident; however, in the interest of product performance, installation jobs PL12E and DST1 contain EDITLIB MOVE directives to force 3TT and its overlays; 1CI and two of its overlays, 1QP, 1BR, and the wideband driver overlays to CM residency. This group of routines and overlays involves 5500 octal words. Based on the site configuration, legitimate directives may be made from EDITLIB comment statements within job deck PL12E to increase CM residency by 1000 octal words.

INTERCOM in an idle state uses 1300 octal words for multiplexer tables and minimum empty buffer chains.

## LCC PROGRAMS

The INTERCOM LCC initializer uses the LCC multiplexer subtables to determine which variants of the LCC programs to load before the LCC driver is brought up. INTERCOM assumes the proper variants are available on the system library and are disk-resident. The following list indicates the names of the LCC programs for which the LCC initializer searches.

<u>LCC Memory Size</u>	<u>8K</u>
Mode 2 ports only defined	0ZD
Mode 3 ports only defined	0ZF
Mode 4 ports only defined	0ZH
Modes 2 and 3 ports defined	0ZJ
Modes 2 and 4 ports defined	0ZL
Modes 3 and 4 ports defined	0ZN

In addition, the LCC autoloading program 0ZA and LCC autodump program 0ZB must also be available on the system library and be disk-resident.

All these programs are available in binary format as part of a separate release of the LCC programs. They can be added to the running system using the following job.

```
Job statement.  
EDITLIB(SYSTEM)  
7/8/9  
READY(SYSTEM,OLD)  
REPLACE(*,INPUT)  
COMPLETE.  
ENDRUN.  
7/8/9  
Binary decks of LCC programs  
6/7/8/9
```

### NPU PROGRAMS

The INTERCOM 2550 Front End NPU initializer uses the NPU multiplexer subtables to determine which variants of the NPU programs to load before the NPU driver is brought up. INTERCOM assumes the proper variants are available on the system library and are disk-resident. The following list indicates the names of the NPU programs for which the NPU initializer searches.

```
Mode 3 ports only defined 0F3  
  
Mode 4 ports only defined 0F4  
  
Modes 3 and 4 ports defined 0F7
```

In addition, the NPU autodump program 0F0 must also be available on the system library and be disk-resident.

All of these programs are available as part of a separate release library for CCP 1 that includes the NPU programs. NPU binaries can be added to the running system using the following job.

```
Job statement.  
EDITLIB(SYSTEM)  
7/8/9  
READY(SYSTEM,OLD)  
REPLACE(*,INPUT)  
COMPLETE.  
ENDRUN.  
7/8/9  
Binary decks of NPU programs 0F0, 0F3, 0F4, 0F7.  
6/7/8/9
```

Section 23 of this part describes CCP 1 installation in detail.

### MUJ SYSTEM ERRORS

INTERCOM multiuser jobs (for example, EDITOR), upon encountering hardware and/or software errors, produce diagnostic dumps. These dumps contain a header MUJ SYSTEM ERROR xx. This message is sent to the system dayfile and to each user currently using the muj. Error codes and their significance are described in the NOS/BE Diagnostic Handbook.

## PASSWORD FILE CREATION

Access to the INTERCOM system is controlled by passwords. The user must specify a valid password to log in to the INTERCOM system. Two types of passwords exist: restricted and unrestricted passwords.

With restricted passwords, when logging in, the user must specify a valid username associated with the given password. The installation defines valid username/password combinations. A user id (two alphanumeric characters) is assigned by the PASSWRD utility, and it is permanently associated with the username/password. This user id is assigned from a pool of available user ids; it is marked as available again only when the username/password is deleted.

With unrestricted passwords, the user may specify any username when logging in. The username is not validated. However, when a user first logs in under a given username, a user id is associated by the LOGIN utility with that username/password combination. Thereafter, this user id is associated with the username/password combination, until the username/password is deleted from the system.

Through the INTERCOM routine PASSWRD, the installation defines valid restricted username/password combinations and valid unrestricted passwords and accounting values to be associated with the username/passwords or passwords. PASSWRD must be called from a data deck submitted to the central site as a batch job. The routine creates two permanent files (or edit existing files). One file, with the permanent file name INTERCOMPASSWORDS, contains all unrestricted passwords, all restricted username/passwords, and all accounting information. The other file, with the permanent file name INTERCOMUNRESTRICTED, contains a bit map defining assigned user ids; it also contains all unrestricted username/password combinations. Installations with many users should do the following.

- Instruct users of unrestricted passwords always to use the same character string for username when logging in.
- Make use, on a regular basis, of the editing facilities in PASSWRD to delete all unrestricted usernames, and so on, freeing user ids.

While a user is in the process of logging in, he is assigned a temporary id. Temporary ids begin with a special character.

The following deck structure can be used to run the PASSWRD routine, creating a password permanent file.

```
Job statement.
PASSWRD.
7/8/9
NEW
ADD
.
.
.
6/7/8/9
```

The following deck structure can be used to modify existing password permanent files.

```
Job statement.
PASSWRD.
7/8/9
OLD
ADD or
DEL
.
.
.
6/7/8/9
```

This mode of PASSWRD operation updates the existing permanent files by adding new or deleting old entries. If both files do not exist, a PF ERROR=12B aborts the run.

To protect against unauthorized modification of the password files, the PASSWRD utility requests permission from the console operator before any modifications are made.

Between the NEW (or OLD) statement and the 6/7/8/9 statement appear the parameter statements which specify the new entries or the editing requirements. After a NEW statement, only ADD parameter statements may appear; after an OLD statement, either ADD or DEL parameter statements may appear. The ADD statement creates a new entry, or replaces an old entry which has the same username/password. The DEL statement deletes one or more entries. The NEW statement may be used to delete existing files entirely and to construct new ones.

The format for an ADD parameter statement is as follows. All numeric parameters must be specified in octal, but the suffix B is not allowed.

ADD U=username,P=password,F=length,T=time,A=acclevl,N=nfiles,E=ecsfl

username	Username (1 to 10 alphanumeric characters) must be specified for restricted passwords; it must be blank or omitted for unrestricted passwords.
password	Password (1 to 10 alphanumeric characters) must be specified. It must be the only unrestricted password of this name defined by the installation. If it is restricted, it must be the only username/password of this particular combination defined by the installation. (If the password or username/password have been previously defined, the ADD card functions as a replace.)
flength	Maximum field length available to the user (1 to 6 octal digits). If blank or omitted, 60000 octal CM words are assumed. This value may not exceed IP.MFL.
time	Time limit for user's session (1 to 4 octal digits, also defines the maximum ETL for individual jobs). If blank or omitted, 500 octal seconds are assumed.
ecsfl	Maximum ECS field length available to the user in multiples of 1000B (1 to 4 octal digits). If blank or omitted, zero is assumed. This value must not exceed IP.MECS.



acclevl      Access level/permission bits for the user (0-3777<sub>8</sub> range). This value defines which programs the user can access. If blank or omitted, an access level of 5 is assumed (dependent on IP.IACES setting in common deck INTCOM).

nfiles        Number of files this user is permitted to attach as local files at any one time (1 to 2 octal digits). If blank or omitted, 24 (octal) files are allowed. This value may not exceed 76<sub>8</sub>.

All parameters start after column 4 on the ADD and DEL statements. They may be specified in any order and should be separated by delimiters (special characters).

The DEL statement is used to delete one or more entries from one or both of the permanent files. It has two formats.

DEL U=username,P=password

DEL I=id

username     May take three forms: 1 to 10 alphanumeric characters, blank, or the character string \*NAMES. If the first form is used, the username/password combination (restricted or unrestricted) is deleted; and the user id becomes available. If the second form is used, all entries in the two files with the given password is deleted. All user ids associated with these entries will become available; the password will no longer be defined. The third form may be used only if the specified password is unrestricted. All entries in the unrestricted password file with the given password will be deleted, and the associated ids will be made available. The password will still be defined.

password     Password to be processed. Whether an unrestricted password is deleted or not depends on the username parameter. If password is \*NAMES, all usernames for all unrestricted passwords are deleted from the permanent files; and the user ids for these usernames become available. The unrestricted passwords will still be defined.

id            User id; may be used as a shorthand notation to specify the username/password associated with this user id. The given username/password entry (restricted or otherwise) is deleted and the user id becomes available. If the password is unrestricted, it will still be defined.

#### SCED INSTALLATION PARAMETERS

When a multiuser job which uses SCED is installed, default parameter values in SCED should be changed to reflect the requirements of the COBOL program involved. A value should be changed by deleting the default definition macro call and replacing it with a call to the SCED macro with the new parameter value. All macros are required.

#### Example

```
*D      SCED.233      Deletes MAXUSR parameter
         MAXUSR 10      Replaces MAXUSR with new value
```

The SCED macro (parameter) calls are described in detail in the INTERCOM 4 Multiuser Job Capability Reference Manual.

<u>Parameter</u>	<u>Default Value</u>	<u>Line to Replace</u>
MAXUSR	30	SCED.233
USAREA	2,214	SCED.234
NUMINT	40	SCED.235
DEFBUF	_____	No parameters No need to replace
OUTBUF	4,45 4,144	SCED.237, SCED.238

#### VERIFICATION PROCEDURE

INTERCOM is brought to control point zero when INTERCOM is entered at the console after the operator has entered the time.

The verification procedure cannot proceed unless a permanent file has been established containing the user passwords.

The following sample from an interactive terminal session indicates if INTERCOM is installed correctly. The underlined characters are typed by the user.

CONTROL DATA INTERCOM

DATE 11/21/75  
TIME 09.27.22.

PLEASE LOGIN  
LOGIN

ENTER USER NAME- THOBBIE

ENTER PASSWORD-

11/21/75 LOGGED IN AT 09.28.46.  
WITH USER-ID D3  
EQUIP/PORT 47/04

COMMAND- SITUATE

USERS WITH SAME PASSWORD

D3-THOBBIE  
OTHERS

B6-HALLA	ER-IPRICE	FL-ALL155
FM-ALL156	BC-OPS	FI-TAYLOR
F3-ZEE	BA-4800BAUD	BB-4801BAUD
BD-HSBT	BE-MSBT	GU-SVLANX
GY-CHESLEY	HN-EBROTH	G4-JGM

BATCH TERMINALS

AS-200UT AU-200UT AV-200UT

AF-XSBT AG-XSBT

COMMAND- ASSETS

ASSETS OF D3 AT 09.30.11.  
EQUIP/PORT 47/04  
FILE QUOTA 20  
FILES IN USE 0  
MAX FL 0077700  
TIME LIMIT 7000  
CP TIME .164  
COMMAND- ETL,100

COMMAND- MAP,ON

COMMAND- ASSETS

ASSETS OF D3 AT 09.31.00.  
EQUIP/PORT 47/04  
FILE QUOTA 20  
FILES IN USE 0  
MAX FL 077700  
TIME LIMIT 7000  
ETL 0100  
MAP ON  
CP TIME .174  
COMMAND- FILES

NONE

COMMAND- LOGOUT

CPA .198 SEC. .198 ADJ.  
SYS TIME 1.159  
CONNECT TIME 0 HRS. 5 MIN.  
11/21/75 LOGGED OUT AT 09.31.49.

RELEASE DESCRIPTION

ALGOL Version 4 operates under the NOS/BE operating system using the minimum hardware configuration as required by NOS/BE.

RELEASE MATERIALS

The program library for ALGOL is contained on release tape PL32.

The release tape contains eight files.

File 1	Program library in UPDATE format
File 2	Compiler relocatable binary
File 3	Compiler absolute binary
File 4	ALGTEXT relocatable binary (COMPASS interface macros)
File 5	Execution-time library relocatable binary.
File 6	Program library in UPDATE format for ALGEDIT
File 7	Relocatable binary of ALGEDIT
File 8	Absolute overlays of ALGEDIT

File 1 is the complete OLDPL common for NOS/BE, SCOPE 2 and NOS. Files 2 through 5, 7 and 8 reflect ALGOL as configured for running under NOS/BE.

LIMITATIONS

The system control statement REDUCE cannot be used since ALGOL programs use the space following the program as buffer and stack areas.

INSTALLATION PARAMETERS

The following installation options are available.

- The type of computer and operating system on which the compiler and its library must be assembled is established on the OLDPL by UPDATE IF DEF directives.

The following directives are necessary to configure ALGOL under NOS/BE.

```
*IDENT MACHINE
*DEFINE COMPUTER6
*DEFINE BUGSAIDA
```

- A macro, DEFAULT, establishes which control error or compilation options are active by default. The release tape contains the following call.

```
DEFAULT      B,L,I,N,E,X
```

To change these values, the parameters must be modified.

```
*DELETE ALG0.119
```

```
DEFAULT new parameters
```

- A macro, DEFAULT, establishes which run time options are to be active by default. No options are default-enabled on the release tape. To set run time default options, introduce directives of the following type.

```
*INSERT OPENALG.102
```

```
DEFAULT new parameters
```

- The default number of significant input characters is set by default to 72, but this value can be changed to a new value.

```
*DELETE ALG0.83  
INPUTLG EQU new value      1<new value<126
```

The K option of the ALGOL control statement may change that value dynamically.

#### INSTALLATION PROCEDURES

Program PL32I produces an updated copy of the eight files of the release tape. It should be used for introducing installation parameters (IDENT MACHINE). Job PL32E may be used to enter the ALGOL 4 binaries into the running system or user libraries.

#### VERIFICATION PROGRAM

Job PL32V can be run to verify the correct installation of ALGOL. The message ALGOL IS INSTALLED should appear on the output file.

---

The information in this section has been deleted.



---

RELEASE DESCRIPTION

The CDC CYBER Cross System executes under NOS/BE to provide support for the CDC CYBER 18 minicomputer and the 2550 series of host communications processors. The CDC CYBER Cross System is composed of the following.

<u>Component</u>	<u>Implementation Language</u>
PASCAL Compiler	PASCAL
Format Program	FORTRAN Extended 4
PASCAL Cross-reference Program	PASCAL
Macro Assembler	COMPASS
Macro File	Assembly
KRONTXT	COMPASS
Micro Assembler	FORTRAN Extended 4
Library Maintenance Program	FORTRAN Extended 4
Link Editor	PASCAL

The CDC CYBER Cross System supports the generation of load modules which may be executed on a CDC CYBER 18 minicomputer or a 2550 communications processor.

HARDWARE CONFIGURATION

The CDC CYBER Cross System requires a minimum of 77000 octal words of central memory for installation and execution. If the 125K PASCAL compiler and 135K PASCAL Cross Reference Program are to be installed, 135000 octal words of central memory are required. With this exception, the minimum configuration is the same as for NOS/BE as described in part II, section I.

RELEASE MATERIALS

CDC CYBER Cross System is released on release tape PL50, the structure of which follows.

<u>File Number</u>	<u>Record Number</u>	<u>File Content</u>		<u>File Type</u>
1	1	UPDATE Program Library		PL
2	1	Format Program	(FRMT)	ABS
2	2	KRONTXT	(KRONTXT)	OVL
2	3	Macro Assembler	(ASSEM)	OVL
2	4	Macro File	(SMAC17)	OVL
2	5	Micro Assembler	(MASSEM)	ABS
2	6	Library Maintenance Program	(MPLIB)	ABS
3	-	(empty)	-	-
4	1	PASCAL Compiler (77K)	(PASCAL)	OVL
5	1	PASCAL Cross Reference Program (77K)	(PASXREF)	DATA
6	1	Link Editor	(MPLINK)	OVL
7	1	Edit	(MPEDIT)	OVL
8	1	PASCAL Compiler (125K)	(PASCAL)	OVL
9	1	6000 PASCAL Compiler	(PASBN01)	OVL
10	1	PASCAL Cross Reference Program (135K)	(PASXREF)	DATA

File 9 contains the binary of the 6000 PASCAL compiler, required for compiling the PASCAL compiler and the Link Editor, and for compiling and executing the PASCAL cross-reference program.

#### PASCAL ORGANIZATION

The PASCAL compiler is organized in a file structure. The components of the compiler are records on the file. The first record of the file acts as a main overlay program and controls loading and execution of the other records. Because of this file structure, the PASCAL compiler cannot be entered into a library via EDITLIB, but must be cataloged as a permanent file (PASCAL). The PASCAL file structure follows.

<u>Record</u>	<u>Name</u>	<u>Function</u>	<u>Implementation Language</u>
1	POSYS	Controls processing	COMPASS
2,3	PASCAL	Compiles programs	PASCAL
4	SYMIO	Performs disk I/O	COMPASS
5	ERRMSS	Table of error messages	Text
6,7	PASDMP	Prints object code listing	PASCAL



## PASCAL COMPILER SYMBOL TABLE PAGE SIZE

The number of entries per symbol table page in the released source of the PASCAL compiler is 1024. Execution field length for this compiler is approximately 77000 octal CM words. For large compilations, increasing the number of entries per symbol table page reduces compilation time dramatically. To generate a PASCAL compiler for installing CCP, the page size is increased to 4096, which causes the compiler to require 125000 octal CM words. Generation of a 125K compiler is accommodated as a CDC CYBER Cross System installation option (refer to Installation Procedure). The minimum recommended page size for any use of the PASCAL compiler is 128. Page size may be altered by changing the compiler source as follows.

```
*DELETE    DPAS1175.1
           PAGESIZE=nnnn.      (nnnn = page size - 1 = 4095 for CCP)
*DELETE    DSYM1175.1
           PAGESIZE EQU nnnn   (nnnn = page size = 4096 for CCP)
*C DPASCAL, SYMIO
```

## PASCAL CROSS REFERENCE PROGRAM TABLE SIZE

If the PASCAL cross reference program is run against a CCP compilation (standard CCP installation jobs do not do so), the following table size modification should be made.

```
*DELETE          XREF.4
           P = 4001
*DELETE          XREF.26,27
OCCURRENCE:     CLASS 30000 OF PACKED RECORD
                LNO: 0..30000
```

This change increases the execution field length requirement for the cross reference program to 135000 octal central memory words. A 135K cross reference program can be generated using a CDC CYBER Cross System installation option (refer to Installation Procedure).

## INSTALLATION PROCEDURE

Job decks PL50I, PL50C, and PL50V may be obtained from the installation deck program library using the procedure outlined in part I, section 1 of this document.

The installation jobs function as follows.

- PL50I Updates the program library with modifications to produce a new program library tape including binary files. If PL50I is extracted from the installation deck program library with CCP defined (refer to part I, section 1), a 125K version of the PASCAL compiler and a 135K version of the cross reference program are produced and written on files 8 and 10, respectively, of the new PL50. The PAGESIZE and XREF modifications, however, are not included on the new program library file (file 1) of PL50. If job PL50I is extracted without CCP defined, 77K versions of the PASCAL compiler and cross reference program are produced and written on files 4 and 5, respectively, of the new PL50. PL50I requires a field length of 135000 octal words to compile the 125K PASCAL and 135K PAXXREF and 77000 octal words to compile the 77K versions. Defining CATALOG causes job PL50I to catalog the new PL50 binaries as permanent files from which they may be executed. The CATALOG option is sensitive to the CCP symbol such that if CCP is defined, a 125K PASCAL and a 135K PAXXREF are cataloged; if CCP is not defined, 77K versions are cataloged.
- PL50C Catalogs the CDC CYBER Cross System binaries from PL50 as permanent files from which they may be executed. Defining CCP causes the 125K PASCAL compiler and 135K cross reference program to be cataloged. Not defining CCP catalogs 77K versions of PASCAL and PAXXREF. If PL50I is run with CATALOG defined, PL50C is not required.

Because the PL50 installation jobs do not enter program binaries into libraries, the DSTn jobs are not applicable.

- PL50V Verifies installation of the CYBER Cross System. It uses the permanent files created either by job PL50I with CATALOG defined or by job PL50C.



RELEASE DESCRIPTION

The CDC CYBER LOADER runs on CDC CYBER 170, CDC CYBER 70 and 6000 Series Computer Systems. CDC CYBER LOADER runs under NOS/BE and requires the same minimum hardware as NOS/BE 1.

RELEASE MATERIALS

The release tape for CDC CYBER LOADER is PL1E which contains a source program library as file 1 and the assembled binary as file 2.

GENERAL DESCRIPTION

PL1E contains LDRTEXT and programs PILOAD, LOADER, LOADU, UCLOAD, LDRCNTL, SEGBILD, FDL.RES, FDL.OCR, FDL.MMI, FOL.RES, SEGRES, TRAP and TRAPPER together with their associated common decks and higher level overlays.

This essentially comprises what is commonly known as the Control-Card-callable Basic Loader, User Call Loader, Fast Dynamic Loader, Overlay Loader, Segment Loader, Fast Overlay Loader, Loader Control Card Processor, and the Debug Aids Package. For further common deck and overlay structure information, consult the LOADER Reference Manual or the IMS.

Note that the program library (PL1E) for the CDC CYBER LOADER contains no Peripheral Processor (PP) programs. These routines are resident on the appropriate system program library PL1B. These routines must exist in the system for correct loader function. The primary loader interface PP programs include the following.

<u>Under NOS/BE 1</u>	<u>Under NOS 1</u>	<u>General Function</u>
LDL	CPM	Read one library directories, loader control word, set protect bit
LDV(LDW)	LDV(LDR)	Perform physical loading Note: Under NOS/BE 1, LDV may call PP program LDW depending on type of load function
LDL	LFM	Assign library files
*	EXU	Load Chippewa binaries (*=NO NOS/BE 1 EQUIVALENT)
ACE	TCS	Advance/crack control statements

In addition, several other peripheral processor programs commonly shared by the entire product set (MEM, MSG, CIO, and so on) are used by the CDC CYBER LOADER.

INSTALLATION PARAMETERS

CDC CYBER LOADER obtains installation parameters from its own local LDRCOM deck. The following installation parameters for the control statement initiated loader may be set at LDRCOM.13 in the update of PL1E. The values shown in parentheses are default values.

IP.PSET (11B)

Core presetting options include the following.

0	Same as 1					
1	Preset to	0000	0000	0000	0000	0000B
2	Preset to	7777	7777	7777	7777	7777B
3	Preset to	1777	0000	0000	0000	0000B
4	Preset to	3777	0000	0000	0000	0000B
5	Preset to	6000	0000	0000	0000	0000B
6	Preset to	4000	0000	0000	00	addr
7	Preset to	2525	2525	2525	2525	2525B
10	Preset to	5252	5252	5252	5252	5252B
11	Preset to	6000	0000	0004	0040	0000B+*

For (6) each location contains its address in the lower 18 bits.

IP.REW (1)

If one, the load file is rewound prior to beginning to load. If zero, no rewind takes place.

IP.LDBG (0) (1)

If nonzero, conditional code to aid in debugging the Loader is assembled. Additional information is available in the Loader IMS.

IP.LDER (1)

Error processing by the loader may be one of the following.

- 0 Abort on all errors (ERR=ALL)
- 1 Abort on fatal errors (ERR=FATAL)
- 2 No abort if possible (ERR=NONE)

IP.FLINC (4000B)

Amount by which field length is increased if loader needs more field length for table construction. May vary up from 100B in increments of 100B.

IP.FLMSG (0) (1)

If nonzero, a dayfile message giving the FL required for loading and execution will be issued for relocatable loads when there is no map.

IP.LRT (0) (1004B)

If nonzero, a dayfile message is issued giving various time and memory measurements. If  $IP.LRT \geq 1000B$ , then the value (IP.LRT-1000B) is placed in bits 29-18 of the MSG call.

IP.PS (60D)

This symbol controls number of lines per page in a load map. It is contained in IPTEXT, not LDRCOM.

IP.PD (6)

This symbol controls the number of lines per inch in a load map. It is contained in IPTEXT, not LDRCOM.

IP.MAP (3) (17B)

Default Loader MAP options include the following.

- |                 |                    |
|-----------------|--------------------|
| 0 = MAP(OFF)    | No map             |
| 3 = MAP(PART)   | S, B options       |
| 13B = MAP(ON)   | S, B, X options    |
| 17B = MAP(FULL) | S, B, E, X options |

- S Loader statistics and error messages only
- B Block names, addresses and lengths
- E Entry point list
- X Cross reference list of external references

## INSTALLATION PROCEDURES

Installation of the CDC CYBER LOADER requires that job decks PL1EI and PL1EE be obtained from the installation deck program library as outlined in part I, section 1.

PL1EI is a maintenance deck which can be used to create a revised program library and binary file. PL1EE can be used to enter CDC CYBER LOADER into the running system or user libraries from either the released PL1E or a tape created by PL1EI. After deck PL1EE has completed, job DST3 can be run to create a deadstart tape of the running system. Job decks PL1EE and DST3 need not be run if the user library installation process is being followed.

---

844-21 and 844-41 Factory Format Support is a software feature that provides FORMAT/FDP and is applicable to NOS/BE running on CDC CYBER 170, CDC CYBER 70, and 6000 Computer Systems that include CEADS/D44 and level A06 (or above) OSY controlware. A08 (or above) OSY controlware is required for 844-41 disk units.

#### RELEASE MATERIALS

The release tape for FORMAT/FDP is PL1F which contains a source program library as file 1 and the assembled binary as file 2.

#### GENERAL AND OPERATIONAL DESCRIPTION

PL1F contains the FORMAT/FDP utility which enables on-line support of 881 and 883 factory formatted disk packs. Factory formatting is the process of preparing a disk surface for use by recording addresses on the disk surface for cylinders, tracks, and sectors. All 881 and 883 disk packs are factory-formatted, surface tested for flaws, and certified for use before shipment to the customer. Under normal conditions, these disk packs remain in use with few further problems. 844-21 and 844-41 Factory Format Support is intended to aid the installation in maintaining disk packs in a usable state and correcting problems that might be encountered.

It is strongly recommended that each installation designate one person as disk pack coordinator, responsible for maintaining all installation disk packs. Only the disk pack coordinator should be allowed to use the pack formatting procedure and the FORMAT utility to do all disk pack formatting. Operators should be forbidden to attempt the operations described herein.

One master aligned disk drive is required for formatting all 881 and 883 packs. As the FORMAT utility tends to monopolize the controller (formatting is actually a controller function), disk pack formatting should be done as a hands-on activity and should not be allowed during a production environment. Operators should be instructed to drop any FORMAT job.

#### FLAWS

All 881 and 883 disk packs are fully surface-tested before being certified for use. Any flaws detected during disk surface analysis are recorded on the disk pack in the utility sector (located at cylinder 410D for 881, cylinder 822D for 883) track 0, sectors 1 and 2 (sector 0 contains the pack serial number).

Surface analysis detects two types of surface abnormalities, hard flaws and soft flaws. Hard flaws are small areas of the disk surface where data cannot be successfully read and written. Soft flaws are those small areas where doubt exists as to the accuracy of repeated reads and writes. To avoid use of these areas, the sector in which they occur is flawed (removed from use) by recording the address (cylinder/track/sector) in the utility sector. The operating system then reads the utility sector and flags known flawed areas as nonusable.

The disk pack coordinator must ensure that all installation disk packs appear flaw-free to the system; that is, all hard and soft flaws must be noted in the utility sector of the disk pack so that the flawed areas will not be accessed by the operating system.

When parity errors are encountered on a disk pack during customer usage, the pack should be returned to the disk pack coordinator for the site, who can then run D44 tests to determine if new flaws have appeared since the last surface test and update the utility sector accordingly. D44 is described in the Concurrent Maintenance Library Reference Manual, which is only available to sites having a Control Data maintenance services agreement.

To keep track of the existing flaws on each disk pack, a disk surface analysis record (DSAR) should be created and maintained for each disk pack in use at the installation.

#### DISK PACK CONDITION

All disk packs in use should have intact factory format information, should have all discovered flaws recorded in the utility sector, and should encounter no parity errors during use.

Disk packs not in use may fall into one of the following categories.

- The factory format information is intact but the content of the utility sector is not accurate; parity errors (recovered or unrecovered) are experienced by the customer. The disk pack coordinator must run D44 tests, compare the output with the DSAR and update both the utility sector and the DSAR. The FORMAT utility, described later in this section, is the only present method for updating the utility sector. When this operation is complete, the pack may be returned to use.
- The factory format data was destroyed or never existed. Customer Engineering must use SMM/FMT utility to initialize factory format sectors. The disk pack coordinator can then run D44 tests. SMM/FMT is run only by Customer Engineering to initialize disk packs at the request of the disk pack coordinator.
- Disk packs with serial numbers below 819683 do not have soft flaws indicated in the utility sector. A disk surface analysis of these packs should be done using D44. All flaws encountered should be entered in the utility sector using the FORMAT utility and noted on the DSAR. The packs may then be returned to use.
- New, repaired, or reconditioned packs should be verified using the FORMAT utility (V) and have a DSAR created before placing the pack in use.

**NOTE**

For both D44 and FORMAT utility operations, it is strongly recommended that only one drive be used to ensure drive-to-drive compatibility of the disk packs. The alignment of this master drive should be checked prior to any utility sector updates.

INSTALLATION INFORMATION

Installation deck PL1FI creates the binaries of FORMAT and FDP. This deck can start execution immediately after the successful installation of SCPTXT in PL1AI. Deck PF1FE enters the assembled binaries produced by PL1FI into either the running system or the appropriate user libraries.

FORMAT UTILITY

The FORMAT utility package is intended solely for the purpose of maintaining 881 and 883 type disk packs for use with the 844-21/844-41 disk drives using 7054/7154/7155 controllers.

- The utility can retrieve the factory recorded manufacturing data, the factory recorded flaw data, and the utility flaw data from a factory formatted disk pack.
- Set/clear sector or track flaws on a factory formatted disk pack.
- Restore the address fields of a previously factory formatted disk pack. (This function is to be used only in the event of loss of addresses on the pack.)

CALL FORMAT AND PARAMETER OPTIONS

FORMAT is a control statement callable CP program that interfaces with a user/operator and a PPU program, FDP, to effect maintenance operations on an 881 or 883 type disk pack that has previously undergone factory formatting. The format of the call card follows.

FORMAT(p1,p2,...,pn)

The parameters are position-independent and may be any of the following.

- I=ifn Defines the input file containing directives and data for controlling utility functions (default is INPUT).
- L=ifn Defines the output file to receive information extracted from the disk pack, and so on. This is the standard output file (default is OUTPUT).
- O=ifn Optional output file in addition to the file specified by L for information retrieved from the disk pack.
- U=xxx Defines the EST ordinal (in octal) of the 844-21 or 844-41 on which the disk pack is mounted.  
Unit must be logically OFF (to the system) and must not contain any active files.  
The U parameter must be specified.

V	Causes the utility to verify the address recorded on the disk pack. V is significant only when MODE = FETCH or MODE = RESTORE.
P=SN	Declares the pack serial number of the pack to be processed. This is a decimal number that should exactly match the serial number recorded on the disk pack at the factory.
MODE=	Declares the operational mode for the utility. Valid declarations follow.
ALTER	Indicates that the input file contains directives to control SET/CLEAR flaw operations.
FETCH	Indicates that the utility is to obtain the information contained on the factory sectors (CYL 410D, TRK 0, SEC 0,1,2 for 881 and CYL 822D, TRK 0, SEC 0,1,2 for 883) and copy it to the output file, and to the optional output file if specified.
RESTORE	Indicates that the utility is to restore addresses and flawed sectors/tracks per the utility flaw map. The utility flaw map must be intact or the program will abort.

Default parameters are equivalent to the following call.

```
FORMAT(I=INPUT,O=0,L=OUTPUT,MODE=FETCH,P=0,U=xxx)
```

The U parameter must be declared to initiate utility processing. The default SN(P=0) always produces an operator message (S/N MISMATCH) and requires a GO.

### INPUT FORMATS

Input to FORMAT contains control directives and flaw data for updating the utility flaw map. Data contained on the input file is examined only when the operational mode has been declared as ALTER; the input file will not be accessed in either the FETCH or the RESTORE modes of operation.

Control directives follow.

SET.	Declares the following data statements contain the addresses of flaws to be set and entered in the utility flaw map.
CLEAR	Declares the following data statements contain the addresses of flaws to be cleared and deleted from the utility flaw map.
FINIS	Declares the end of the input data. No information following this card will be processed by FORMAT. This directive is optional.

Data cards are of the following format.

```
x,cccc,tt,ss
```

x	S or T, to indicate a sector or a track flaw.
cccc	Octal number specifying the cylinder (0 - 632B for 881, and 0-1466B for 883).
tt	Octal number specifying the track (0 - 22B)
ss	Octal number specifying the sector (0 - 27B). ss field is ignored for track flaws.

All input data is checked to ensure that values are within range. Any errors in the input result in termination of the utility prior to accessing the disk. SET and CLEAR directives can be intermixed in the input; however, all CLEAR operations are performed before any SET operations. Any attempt to alter the status of the factory sectors results in an error. All control directives and data start in column 1. A maximum of 157 data statements can appear in the input stream.

### OUTPUT FORMATS

Output generated by FORMAT always goes to the standard output file. Additionally, output generated as a result of a FETCH operation can be directed to a second file; this file can then be used as input to another program or disposed to either card or hard copy, and so on. Format of data in the optional output file is identical with input formats, however, no directive cards are used.

For all modes of operation, standard output contains the following information.

- Listing of the input stream, if any.
- Pack serial number and data of factory formatting from the manufacturing sector (CYL 410D, TRK 0, SEC 0).
- Listing of the factory flaw map as continued on CYL 410D, TRK 0, SEC 1 for 881; CYL 822D, TRK 0, SEC 1 for 883.
- Listing of the utility flaw map as contained on CYL 410D, TRK 0, SEC 2 for 881; CYL 822D, TRK 0, SEC 2 for 883.
- Listing of the utility flaw map following any changes resulting from SET and/or CLEAR directives. (MODE = ALTER only).
- Listing of flawed sectors and tracks as read from the disk during address verification (MODE = FETCH or MODE = RESTORE).

#### OPERATOR INTERVENTION AND CONSOLE MESSAGES

Operator intervention is required on all ALTER and RESTORE operations as a safeguard against accidental pack destruction. In addition, if the pack serial number parameter does not match the serial number recorded on the disk pack, the operator is given the option of dropping the job or overriding the condition and allowing the job to run.

The following console messages are displayed to inform the operator of the status of the function being performed or the need for intervention to continue processing.

ALTERING FLAW MAP S/N=xxxxxx; status message indicating utility flaw map is undergoing modification.

RESTORING ADDRESSES S/N=xxxxxx; status message indicating pack is currently undergoing restoration of address fields. Control point should not be dropped while message is displayed.

FETCHING FLAW DATA S/N=xxxxxx; status message indicating factory recorded data is being retrieved from CYL 410D, TRK 0, SEC 0,1,2 for 881; or from CYL 822D, TRK 0, SEC 0,1,2 for 883.

VERIFYING ADDRESSES S/N=xxxxxx; status message indicating read-only pass is being made across pack. Message is displayed after successfully fetching factory-recorded data and flaw maps or successfully restoring address fields if VERIFY option (V) was specified on program call card.

S/N MISMATCH - xxxxxx GO/DROP; flashed when P parameter is not identical to serial number found on pack. Operator must intervene to continue processing.

xxxxxx TO BE ALTERED GO/DROP; flashed whenever utility flaw map is to be modified. (MODE = ALTER). Operator must intervene to continue processing.

xxxxxx TO BE RESTORED GO/DROP; flashed whenever address fields are to be rewritten, (MODE = RESTORE). Operator must intervene to continue processing.

In all the preceding messages, xxxxxx signifies the serial number as read from the manufacturing data recorded in CYL 410D, TRK 0, SEC 0 for 881; and CYL 822D, TRK 0, SEC 0 for 883.

#### DAYFILE MESSAGES

In addition to the console messages, which are entered in the system and control point dayfiles, the following messages are entered in the dayfiles to record catastrophic conditions that caused the program to abort.

EST ORDINAL xxx INVALID OR UNAVAILABLE; indicates EST. ordinal xxx, defined by U=xxx on call card, is unusable. EST entry is printed in octal in output.

FILE EQUIVALENCE MAY NOT BE 0; indicates either input file or standard output file has been declared empty.

ILLEGAL FILE NAME - xxxxxx; indicates that file has been given an illegal name.

INVALID DATA IN INPUT STREAM; indicates that input file contains incorrect data. Refer to input stream listing for card in error.

INVALID PARAMETER ON PROGRAM CALL CARD; indicates at least one unrecognized or ill-formed parameter found.



MANUFACTURING DATA INVALID; indicates that one of the factory-recorded sectors containing either manufacturing or flaw data is either unreadable or not present. Refer to output for detailed status indicating actual problem. If factory-recorded data cannot be read, pack may not be processed using this utility.

SERIAL NUMBER MUST MATCH ON ALTER; indicates attempt made to modify utility flaw map without first obtaining exact match between P parameter and serial number recorded on the pack. Since this may result in destruction of valid data, override is disallowed. Refer to output listing for actual serial number read.

TABLE OVERFLOW ON INPUT; this message indicates that too many FLAW statements were found in the input stream. FLAW input limit is 157B flaws.

UNRECOVERABLE ERROR CONDITION OCCURRED; indicates utility operation was terminated because of nonrecoverable error. Refer to general and detailed status in output listing for specific error condition. If this condition occurs, it is extremely probable that pack and/or drive is unusable in its present condition.

#### INSTALLATION PROCEDURES

Installation job decks PLIFI and PLIFE can be obtained from the installation deck program library using the procedure outlined in part I, section 1.

PLIFI is a maintenance deck which can be used to create a revised program library and binary file. Job PLIFE can be used to enter FORMAT/FDP into the running system, after which job DST3 may be run to create a deadstart tape of the running system. If the user library installation process is being followed, PLIFE and DST3 need not be run.



---

### GENERAL DESCRIPTION

CDC CYBER Database Control System (CDCS) is a partial implementation of the concepts embodied in the Data Base Facility Proposal, March 1973, and DDL Journal of Development, January 1974.

COBOL conventional I-O verbs operate through CDCS to provide file linkage through related items, data validation, data conversion, data independence (through DDL Schema and Sub-Schema descriptions), record restructure, data base procedure execution, and record logging. CDCS performs I-O through CRM on basic access method files and advanced access methods files. CDCS record logging is performed by routines in Data Base Utilities (DBU). Consequently, DBU must be installed in order to run a CDCS job with logging. Refer to section 22 for information on the installation of DBU.

### RELEASE DESCRIPTION

CDCS runs under the NOS/BE operating system. CDCS requires the same minimum hardware configuration as NOS/BE. CDCS will typically add between 20K (octal) and 34K (octal) to the field length required to load and between 10K (octal) and 22K (octal) to the field length required to execute after CDCS initialization.

### RELEASE MATERIALS

CDCS is released on program library tape PL54. The structure of the release tape includes the following.

File 1	Program library
File 2	CDCS binary, relocatable format
File 3	CDCSTXT, system text file for error diagnostics

### INSTALLATION PROCEDURE

Installation decks PL54I, PL54E, and PL54V can be obtained from the Installation Deck program library using the procedure outlined in part I, section 1.

Deck PL54I serves as a program library maintenance deck in that it allows regeneration of the CDCS program library and binary files. Deck PL54E will use EDITLIB to enter CDCS into the running system or user libraries either from the release tape or from a tape created by deck PL54I.



RELEASE MATERIALS

QUERY UPDATE is released on the program library tape PL55. The structure of the release tape is as follows.

- File 1 Program library
- File 2 QUERY UPDATE, and REPORT absolute overlays
- File 3 Absolute binaries of QUERY UPDATE, and REPORT
- File 4 Owncode linkage module; binary, relocatable format

HARDWARE CONFIGURATION

QUERY UPDATE requires the same minimum hardware configuration as NOS/BE. A minimum of 30K octal words of central memory is required to execute this product. A typical minimum job requires approximately 5K octal more for buffers in order to run.

INSTALLATION PARAMETERS

The common deck IPARAMS, present in system text IPTEXT, does not contain any parameter specific to QUERY UPDATE. IPARAMS is used to test for the installed character set (IP.CSET) and for the format of the system date (IP.YMD). Part III of this document contains a cross-reference map of QUERY UPDATE routines versus symbols in IPARAMS.

Assembly options are defined within QUERY UPDATE. At the time of release, the assembly options are set to values deemed most convenient or practical. For example, default report page size is 136 columns x 60 lines. To obtain an up-to-date listing of the assembly options, run a job containing the following control statements and directives (the program library for QUERY UPDATE should be available on file OLDPL).

- UPDATE,Q,L=0.
  - COPYSBF,COMPILE,OUTPUT.
  - 7/8/9
  - \*IDENT CPT
  - \*BEFORE CWEOR1.1
  - \*DECK OPT
  - \*CALL OPTIONS
  - \*C OPT,CWEOR1
- Common deck containing  
the assembly options

NOTES AND CAUTIONS

QU 3 requires either DDL 2 or DDL 3 to run. DDL 2 is released on PL56; DDL 3 is released on PL77. The installation deck expects to find the syntax table generator SYNGEN in PL56 or PL77, depending on the =DEFINE used in PL55I. Installation of QU 3 requires defining DMGMNT in jobs DST1, DST2, and DST3.

INSTALLATION REQUIREMENTS

1. Before installing QUERY UPDATE, the following products must be installed.
  - CDC CYBER Record Manager (AAM) Including the AAM indexed sequential access method (IS), the direct access method (DA), the actual key access method (AK), the multiple index processor (MIP). The access method IS is mandatory; DA/AK/MIP are optional if the corresponding features are not used.
  - SYMPL Version 1
  - Sort/Merge Version 4 If the directive SORT is to be used  
  
If SORT or one of the access methods are not installed, missing entry points show in the load map when QUERY UPDATE is installed.
  - DBU Version 1 If logging is to be used
2. In addition, SYNGEN, a special syntax table generator, is needed to compile QUERY UPDATE. The installation deck expects SYNGEN to be found in PL56 or PL77 as part of file 1.

## INSTALLATION PROCEDURE

Installation job decks PL55I, PL55E, PL55O and PL55V can be obtained from the installation deck program library using the procedure outlined in part I, section 1.

PL55I would be used to modify the PL, build a NEWPL, assemble and compile the entire QUERY UPDATE PL, and generate and save the relocatable and absolute binaries on the NEWPL.

PL55E should be used to install QUERY UPDATE into the running system or user libraries.

PL55O, applicable only in the user library method of installation, allows regeneration and replacement of absolute overlays in the user library plus creation of a new PL55 tape.

The main overlays of QUERY UPDATE and REPORT are placed in the library NUCLEUS, the other overlays in the library SYSOVL. At the time QU 3 is edited into the system, an EDITLIB directive is supplied to delete overlay 4-0 of QU 2, since QU 3 has no corresponding overlay. If QU 2 was not present in the first place, the nonfatal error condition that will occur can be safely ignored. The Data Base Procedure linkage module is placed in SYSMISC.

QUERY UPDATE is able to produce reports using the DESCRIBE directive even if DDL version 2 or 3 is not installed. Without DDL Version 2 or 3, the USE, EXTRACT and other file manipulation directives cannot be processed.

On-site modifications to the PL55I job are required if neither DDL2 nor DDL3 are installed. If DL2 is defined when extracting the PL55I job, the new PL56 tape is requested by the job; otherwise, the new PL77 tape is requested.

---

RELEASE DESCRIPTION

DDL is released on the release tape known as PL56. The structure of the release tape is as follows.

File 1	Program library (including SYNGEN)
File 2	DDL binary; absolute format
File 3	Directory access routines; relocatable format
File 4	DDL binary; relocatable format

HARDWARE CONFIGURATION

DDL requires the same minimum hardware configuration as NOS/BE. A minimum of 50K CM is required to execute DDL.

INSTALLATION PROCEDURE

Installation job decks PL56I, PL56E, PL56O, and PL56V can be obtained from the installation decks PL using the procedure outlined in part I, section 1.

PL56I can be used to modify the PL, build a NEWPL, assemble and compile the entire DDL PL, and generate and save the relocatable and absolute binaries on the NEWPL. In non-ULIB mode, PL56I executes an EDITLIB to the running system DMSLIB library to make the directory access routines available for products that use the CDCS interface.

PL56E is used to install DDL into the running system or user libraries.

Job PL56O allows regeneration and replacement of absolute overlays on a new PL56 tape.

The main overlay of DDL is placed in the library NUCLEUS, the other overlays in the library SYSOVL. The directory access routines are placed on the library DMSLIB.

SYNGEN, a special syntax table generator, is needed to compile DDL. SYNGEN, now on PL56, is designed to facilitate the implementation of syntax driven software. It is required to compile DDL and QU 3.





RELEASE MATERIALS

BASIC 3 is released on tape PL57. PL57 contains three files. File 1 contains the source code for BASIC 3 in Update program library format; file 2 contains the absolute binary of the compiler; file 3 contains the relocatable binaries of the runtime library routines.

HARDWARE CONFIGURATION

The minimum configuration to operate BASIC 3 in batch mode is the same as the NOS/BE minimal configuration. The minimal configuration to operate BASIC 3 interactively is the same as the minimal configuration for INTERCOM under NOS/BE.

NOTES AND CAUTIONS

Because dynamically allocated strings cannot be implemented without changing the object code generated by BASIC, the object code generated by BASIC 3.1 is not fully compatible with that generated by later versions of BASIC. BASIC 3.1 relocatable binaries will not execute under the later version BASIC library. BASIC 3.1 programs maintained in object form must be recompiled under the later version of BASIC before they can be executed under the later versions. However, no source code changes or conversions are required.

INSTALLATION PARAMETERS

IPARAM symbol IP.CSET is used to control which character set, 63 or 64, BASIC is assembled to support.

The COMPASS micro MODLEVEL is used to specify the level of the BASIC compiler output in generated relocatable binary decks. The value of MODLEVEL is controlled by the ML parameter on the COMPASS control statement. Refer to the COMPASS Reference Manual for details.

There are six installation options controlled by BASIC symbols. Their release settings and Update modifications required to change them are as follows.

PD (print density) default (release value=6 lines/inch)

\*DELETE LIPARAM.2

IP.PD CEQU 8 PD DEFAULT=8 LINES/INCH

PS (page size) default (release value=60 printable lines/page)

\*DELETE LIPARAM.3

IP.PS CEQU n PS DEFAULT IS n LINES/PAGE

n Any value equal to or greater than 4.

BL (burstable listing) default (release value=0; that is, listing is not burstable)

\*DELETE LIPARAM.4

IP.BL CEQU 1 BL DEFAULT=1 (BURSTABLE)

AS (ASCII parameter) default (release value=0; that is, not ASCII)

\*DELETE LIPARAM.5

IP.AS CEQU 1 AS DEFAULT=1 (ASSUME ASCII)

Array base default (release value=1)

\*DELETE BASCOMP.202

BDFLT DATA 0 DEFAULT ARRAY ORIGIN=0

Messages giving time/memory required to compile/execute (release value is 0; that is , messages are off)

\*DELETE LIPARAM.9

MESSAG EQU 1 TURNS ON MESSAGES

**NOTE**

If IPARAM symbols are defined for print density, page size, burstable listing, and ASCII options, they will override the conditional EQUs for BASIC.

INSTALLATION PROCEDURES

BASIC is conditionally assembled for either NOS or NOS/BE. The IPARAM symbol OS.NAME must be SCOPE in order to generate the NOS/BE variant of the compiler.

Installation decks PL57I, PL57E, and PL57V may be obtained from the Installation Decks PL using the procedure described in part 1, section 1.

Job PL57I is a maintenance deck which can be used to create a revised release format tape containing a modified program library and assembled binary. Job PL57E can be used to enter BASIC into the running system or user libraries through EDITLIB, either from the released tape or from the tape created by deck PL57I.

VERIFICATION PROGRAM

The verification program supplied with the release (PL57V) compiles and executes two BASIC programs in batch mode. The first verifies that the compiler has been installed correctly and the second that the relocatable version of the runtime system has been installed correctly.

Less than one minute is required to run the verification program deck after BASIC has been installed.

---

### RELEASE DESCRIPTION

Data Base Utilities (DBU) run under the NOS/BE operating system in conjunction with CDCS and QU to create a log file of changes to a data base and to recover or restore the data base.

The logging part of DBU runs in the field length with a CDCS job and adds 2000 (octal) words (plus CRM buffers) to its field length. DBU logging is added to an absolute overlay in QU.

The minimum hardware configuration for recover (entry point DFRCV) and restore (entry point DFRST) is the minimum required by the operating system. Minimum execution field length is 62K (octal). Typical is 75K (octal).

### RELEASE MATERIALS

DBU is released on program library tape PL58. The structure of the release tape is as follows.

File 1	Program library
File 2	Logging binary, relocatable format
File 3	Recover/restore binary, relocatable format
File 4	Recover/restore binary, absolute format

### INSTALLATION PROCEDURE

Installation decks PL58I, PL58E, and PL58V can be obtained from the installation decks PL using the procedure outlined in part I, section 1.

Deck PL58I serves as a program library maintenance deck in that it allows regeneration of the DBU program library and binary files. Deck PL58E uses EDITLIB to enter DBU into the running system or user libraries either from the release tape or from a tape created by deck PL58I.

### LIMITATIONS

Installation of DFRCV (recover) and DFRST (restore) requires that Sort/Merge and the FORTRAN interface to Sort/Merge be installed.



RELEASE DESCRIPTION

Version 1 of the Communication Control Program (CCP) is the software and loadable controlware that supports the 2550 Host Communications Processor (HCP) as a front end to INTERCOM 4 on CDC CYBER 170, CDC CYBER 70, and 6000 series computers. The CCP binary load modules reside in the NOS/BE operating system PPU library so they can be loaded into the 2550 by INTERCOM.

The release tape consists of the CCP source program file (which includes Post Link Editor initialization directives), Multiplex Subsystem (MSS) object text file, two binary load files, two intermediate files, and a print file.

CCP installation creates a CCP downline load module that resembles a PPU binary. The load module is named 0F3, 0F4, or 0F7 and is installed via EDITLIB in the operating system. To create the load module, the following procedures are necessary.

1. Create permanent files which contain the CCP source program file, MSS controlware object text file, and MPEDIT object text file.
2. Generate a local load module. The CCP 1 release tape does not contain the utilities needed to generate the load module. These utilities are provided by the CDC CYBER Cross System described in that section; they are programs that execute in the operating system environment, producing object code that executes in the 2550. The CDC CYBER Cross System consists of a PASCAL compiler, MACRO assembler, MICRO assembler, Link Editor, and Post Edit program. The CCP local load module is created by processing the CCP source file and MSS object file through the CDC CYBER Cross System. The processing of these files consists of compilation, assembly, and link editing that results in a local load module (refer to CDC CYBER Cross System General Information Manual).
3. Generate a downline load module. The CCP local load module file and Post Link Editor initialization directives are used as inputs to the Post Link Editor, MPEDIT, which initializes CCP tables and generates the CCP downline load module. Up to three CCP downline load modules may be created. 0F3 supports mode 3 (TTY) terminals, 0F4 supports mode 4 (200 UT) terminals, and 0F7 supports both mode 3 and mode 4 terminals.

Install each CCP downline load module via EDITLIB.

The Bootstrap Dump program must also be generated and installed as a downline load module. It appears as a PPU binary named 0F0 when it has been installed.

The following description provides information needed to install CCP on the NOS/BE operating system. This section assumes the CDC CYBER Cross System package has been installed as described in that section. Information needed for source code modification is included.

RELEASE MATERIALS

Release materials consist of one 7- or 9-track system standard label magnetic tape. The PL61 tape has the following contents.

<u>File Number</u>	<u>File</u>	<u>Format</u>
1	CCP source program file	UPDATE sequential program (including MPEDIT library format initialization directives)
2	MSS loadable firmware	CDC CYBER Cross System object object textlibrary file format
3	Dump Bootstrap	Downline load file 0F0
4	Downline load file	Downline load file 0F7
5	ABSOLMP	Output from MPLINK
6	SYMTAB	Symbol table output from MPLINK
7	Print	ASSEM and PASCAL listing of 0F7

## HARDWARE CONFIGURATION

The minimum hardware configuration to build CCP 1 requires a job field length of 77,000 octal words while running NOS/BE. (Running PASCAL with a field length of 77K requires a long time to run on a busy system; using the 125K version is much faster.)

### CCP HARDWARE REQUIREMENTS

The minimum equipment configuration required to execute the CCP consists of the following.

- 1 2550-2 Host Communication Processor includes:
  - 1 Multiplexer Loop Interface Adapter
  - 1 Loop Multiplexer
  - 1 Cyclic Encoder board
  - 1 CDC CYBER Communications Coupler
- 1 32K memory unit with 2550-2 processor
- 1 Communications Line Adapter from any of the following.
  - 2560-1 Synchronous CLA
  - 2561-1 Asynchronous CLA

### NOTES AND CAUTIONS

The communications line adapter slots in the loop multiplexer should be assigned in order of decreasing line transmission speeds. For example,

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	

### INSTALLATION PARAMETERS

The following types of parameters can be adjusted during the creation of CCP software load files.

1. MPEDIT constants
2. CCP PASCAL source file constants
3. UPDATE DEFINE directives used during compile file creation
4. CCP or BOOTDUMP macro assembler source constants

The following charts list the installation parameters, together with acceptable and default values. Numeric values preceded with a \$ are given in hexadecimal.

All statements for types 1 and 2 end with a semicolon and can be followed by a comment which is preceded by a right arrow and followed by a down arrow.

The syntax for items of type 1 and 2 is as follows.

```
name = value;    comments
```

There are no column restrictions.

The syntax for items of type 4 is as follows.

```
EQU    name(value)    comments
```

The following are MPEDIT constants. Delete the desired UPDATE identifier and insert the corresponding name and associated value.

<u>UPDATE Identifier</u>	<u>Name</u>	<u>Description</u>	<u>Acceptable Values</u>	<u>Default Value</u>	<u>Restrictions</u>
ZD10.232	/JICORESIZ	Core memory size of NPU	/CS32K /CS48K /CS64K	/CS48K	Core size must be less than or equal to the actual physical core size.
ZD10.212	/C4LCBS	Number of lines that can be configured	1 - 127	127	Memory space must be large enough to accommodate. Refer to Configuration Aid 1. CLA addresses cannot be greater than C4LCBS. Must be equal to installation parameter C4LCBS.
CC10038.16	/B0MT1	Minimum number of small data buffers	1 - n Refer to Configuration Aid 2	50	
ZD10.185	/MD4IBL	Maximum mode 4 input block length (characters)	1 - n	1280	Reference INTERCOM IP.IMXL4 parameter
CC10046.492	/TTYIBL	Maximum TTY input block length (characters)	1 - n	100	Reference INTERCOM IP.IMXL3 parameter
ZD10.205	/C6CCC	Coupler connect code	\$C, \$D	\$C	\$C for primary coupler \$D for expansion coupler
CYCLE NO. 1	/CYCLE	Source file cycle number (for identification purposes only)	0 - \$FF	\$1	
CC10022.116	G0LFSTRIP	If true, extra line feeds on TTYs are suppressed. Card format is G0LFSTRIP :=/TRUE;	/TRUE /FALSE	/FALSE	Requires PSR IN41615 in INTERCOM

The following is a CCP PASCAL source file constant.

<u>Card Identifier</u>	<u>Name</u>	<u>Description</u>	<u>Acceptable Values</u>	<u>Default Value</u>	<u>Restrictions</u>
CONST.26	C4LCBS	Maximum number of lines that can be configured	1-127	127	Core size must be able to accommodate number of lines specified. Must be the same as installation parameter /C4LCBS. Must be greater than or equal to number of lines defined in INTERCOM mux subtable.

The following is a CCP or BOOTDUMP macro assembler source constant.

<u>Card Identifier</u>	<u>Name</u>	<u>Description</u>	<u>Acceptable Values</u>	<u>Default Value</u>	<u>Comments</u>
CC10101X.6	CEQUIP	Coupler connect code for BOOTDUMP	\$C, \$D	\$C	\$C for primary coupler. \$D for secondary coupler. Must be the same as installation parameter ZD10.205 in MPEDIT constants.

The following UPDATE DEFINE names are used during the UPDATE that produces the CCP compile file. The DEFINES select CCP software modules. These DEFINE directives must be specified for compile file generation of both CCP source and MPEDIT directives. Refer to the UPDATE Reference Manual for further information.

<u>Name</u>	<u>Description</u>	<u>Acceptable Values</u>	<u>Default Value</u>	
DEBUGALL	If defined, build in Debug Aids	DEBUGALL or omitted Refer to Configuration Aid 1	DEBUGALL	(In installation deck PL611)
NOTE				
DEBUGALL must be defined if Test Utilities Package (TUP) is to be used.				
STAMPING	If defined, buffer stamping (another debug aid) will be invoked	STAMPING or omitted Refer to Configuration Aid 1	omitted	
TTY	If defined, the TTY Terminal Interface Package (TIP) will be included for terminal types 1, 2, 3 and 4	TTY or omitted Refer to Configuration Aid 1	TTY	
MODE4	If defined, the MODE4 TIP will be included for terminal types 5, 6, 7, and 8	MODE4 or omitted Refer to Configuration Aid 1	MODE4	

### CONFIGURATION AIDS

#### MEMORY SPACE REQUIREMENTS

The following lists the 2550 memory space required for available software.

<u>Feature</u>	<u>Words (Decimal)</u>
Basic software (required)	25000
Debug aids	2400
TTY Terminal Driver	2400
Mode 4 (terminal driver)	3600

If buffer stamping is invoked, one word must be set aside for each potential buffer of the smallest allocated size. Since the smallest buffer size is eight words, an upper bound on the memory space required for buffer stamping can be found with the following formula.

memory required for buffer stamping =

$$(\text{memsize} - (10752 + \text{tip} + \text{debug})) / 9$$

where

memsize	32768 if 32K version is installed 49152 if 48K version is installed 65536 if 64K version is installed
tip	1280 if TTY terminal driver is installed 1920 if mode 4 terminal driver is installed
debug	1280 if debug aids are installed 0 otherwise



The space left over is used for line tables, terminal tables, and dynamic buffer allocation. Guidelines for the utilization of this space are the following.

<u>Type</u>	<u>Words per Line</u>	<u>Words per Terminal</u>
Teletypewriter	42	65
Mode 4 interactive	42	75
Mode 4 batch	42	830

The above figures are approximations only; exact memory utilization is a function of block size, line speed, and so on.

**NOTE**

Do not configure a 2550 for more than it can accommodate, or serious degradation of throughput will occur. Observe that the 2550 configuration is specified via both CCP 1 and INTERCOM installation parameters.

**BUFFER POOL SELECTION**

Buffer pools must be specified correctly to accommodate terminal tables, work lists, system protocol messages and data without forcing extreme amounts of dynamic pool adjustments.

Buffer pools should be specified using the following notation.

- A. Maximum number of TTY terminals active at one time
- B. Maximum number of terminals configured
- C. Number of buffers (at 26 characters per buffer) required for the average TTY message

Small data buffer pool = (A x C) or B, whichever is larger, + 20.

**INSTALLATION PROCEDURES**

The NOS/BE installation deck program library contains five decks to be used in the installation of CCP 1. In addition, deck PL501 and/or PL50C must be run to install the CDC CYBER Cross System as permanent files to be used in generating CCP 1.

PL6111 This deck updates the CCP 1 source library and makes the NEWPL a permanent file. It then copies the remaining files to disk and makes them permanent files to be used by other jobs in the build process. Installation parameter changes or source code changes should be made here.

Before the following jobs are run, PL6111 must be run.

PL6112 This job builds a new bootstrap dump binary to replace the one cataloged by job PL6111.

PL6113 This job builds new CCP 1 downline load modules. The following definitions apply when extracting this job from the installation deck program library.

- 0F3 Build the TTY only version of CCP
- 0F4 Build the MODE4 only version of CCP
- 0F7 Build the TTY and MODE4 version of CCP
- CCP64 Build the 64K version of CCP 1

/JICORESIZ must be changed also

PL6112 and PL6113 may be run concurrently. It is necessary to run these jobs only if there are changes to the appropriate module.

This job is set up to run with a field length of 125K. If it is necessary to run in 77K, the RFLs must be changed.

The downline load file on the release tape contains the 48K 0F7 version of CCP. When the new PL61 is rewritten in job PL6114, the downline load file, ABSOLMP, SYMTAB, and PRINT is replaced with the version built in job PL6113. If more than one version of CCP 1 is to be maintained, job PL6113 must be run again for each version, making appropriate changes to permanent file names.

PL61I4 This job creates a new PL61 from the permanent files created by the previous jobs. If ULIB is defined during the extraction of this job, it then enters the bootstrap dump (PPU0F0) and the downline load file into a user library called CCPLIB for subsequent capture by DST1.

PL61E This job enters PPU0F0 and the downline load module from the PL61 tape into the system PP library. DST2 or DST3 can be used to capture the resultant system.

#### CORRECTIVE CODE

When corrective code for CCP 1 is issued, it is necessary to generate a new system. The method of incorporation of these updates into the CCP is dependent on the type of update. The updates can be categorized as follows.

Source updates (including MPEDIT Initialization directive updates)

MPEDIT Initialization Directive updates only

Modifications to CCP source programs are in Update format. The corrective code should be added to its associated Update deck. After the modifications have been incorporated into the source deck and a new source file created, an updated system can be generated using the decks PL61I2 and PL61I3.

#### VERIFICATION PROGRAMS

The verification of the CCP can be divided into the verification of system generation and the verification of the online system.

#### SYSTEM GENERATION

In order for the 2550 system to run properly, the CCP must complete the system generation procedures error free. Each phase of building the system must finish processing with no errors before the next procedure is initiated. The NOS/BE 1 programs which can detect errors during system generation include Update, MACRO Assembler, PASCAL Compiler, Library Maintenance, Link Editor, and Post Edit Program. The following reference manuals should be consulted for the identification and explanation of specific types of errors.

Update Reference Manual

CYBER Cross System 1  
MACRO Assembler Reference Manual

CYBER Cross System 1  
PASCAL Compiler Reference Manual

CYBER Cross System Version 1 Link Editor and  
Library Maintenance Programs Reference Manual

As released, the CCP should complete system generation without errors. If the installation parameters are modified with care and the restrictions on them adhered to, errors should not occur while building the CCP.

#### ON-LINE SYSTEM

Refer to the INTERCOM verification procedure.

RELEASE MATERIALS

COBOL version 5 release material consists of a magnetic tape identified as PL60. The structure of PL60 is as follows.

File 1	Update program library of the compiler and object routines
File 2	Relocatable binary records of the compiler
File 3	Absolute binary records of the compiler texts
File 4	Relocatable binary records of the object-time routines
File 5	Termination capsule binaries
File 6	Termination dump relocatable binaries
File 7	Absolute binary records of the compiler that may be installed on the system or onto a user library
File 8	Absolute binary record of the termination capsules
File 9	Termination dump absolute binaries

LIMITATIONS

The ANSI Communications Facility is not available.

Most user programs written for COBOL 4 require translation before they will compile and execute properly under COBOL 5. A conversion aid is available. Installation of this conversion aid is described in section 25. Refer to the COBOL 4 to COBOL 5 Conversion Aid Reference Manual for a full description of this product.

The support of 63-character collating sequences is achieved by replacing the collating character % (octal value 63) with the character : (octal value 00), when the installation selects the 63-character set option. The pseudo-names CDC-64 and ASCII-64 in the ALPHABET clause then refer to the CDC-63 and ASCII-63 collating sequences, respectively.

INSTALLATION PARAMETERS

CB5TEXT selects symbol definitions from IPTEXT for use by the COBOL 5 compiler. There are no direct references to any IPTEXT symbols within the compiler or object routines, thus allowing the installing site greater flexibility in changing their normal installation parameters for COBOL 5.

Symbols governing machine type, character set, and CMU option are obtained from IPTEXT, while those governing CDCS, the default page size and print density, and the default error termination must be changed within the product.

To override the system defaults in these areas, make the following changes.

To generate a compiler that will generate code for a CMU machine, insert the following after the \*OPTION= and before the OP.BDP label in the CB5TEXT deck.

```
OP.BDP  CEQU  OP.YES
```

To change the default error termination level to T, W, F, or C, use 1, 2, 3, or 4, respectively. Change the DEF CB5\$ET statement in the ASSEMOP deck to

```
DEF      CB5$ET      level ;
```

To activate CDCSI processing, change the statement with label OP.DCS in deck CB5TEXT to the following.

```
OP.DCS  CEQU  OP.DCSI
```

Also change the DEF CB5\$CDCS statement in deck ASSEMOP.

```
DEF      CB5$CDCS    = #CDCSI # = ;
```

Both of the above changes must be made or the results are unpredictable. These changes are activated if DMGMNT is =DEFINED during the extraction of installation job PL60I.

To activate CDCS2 processing, change the statement with label OP.DCS in deck CB5TEXT to the following.

```
OP.DCS CEQU OP.DCS2
```

Also change the DEF CB5\$CDCS statement in deck ASSEMOP.

```
DEF CB5$CDCS ≡≠CDCS2≠≡;
```

Both of the preceding changes must be made or the results are unpredictable. These changes are activated if DMGMNT and CD2 are =DEFINED during the extraction of installation job PL60I.

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 control statement; the next in order is the installation-specified value of CB5\$PDENS (if other than zero); and the last is the value of IP.PD in IPTEXT.

To select a default print density different from that specified in IPTEXT, change the line for CB5\$PDENS in ASSEMOP, using n= 3, 4, 6, or 8 lines per inch, as follows.

```
DEF CB5$PDENS ≡ n≡
```

The number of lines per page is determined by one of the following, in descending order of dominance. The dominant factor is the PS parameter on the COBOL5 control statement; the next in order is the installation-specified value of CB5\$LINP (if other than zero); and the last is the result of the following calculation.

Lines per page = Print density\*(IP.PS/IP.PD)

Print density in the calculation is the density determined from the factors described previously.

If a change to CB5\$LINP is desired, locate CB5\$LINP in ASSEMOP and change it to the following.

```
DEF CB5$LINP ≡ n≡
```

The value n is an integer. The page will contain n lines, including three lines at the top and three lines at the bottom for headings.

To change the CPU type that code is generated for (and object routines are assembled for); it may be set to OP.6400 for a machine with a unified CPU or to OP.6600 for a machine with a nonunified CPU; insert the following statement after the \*OPTION= statement and before the OP.MODEL label in deck CB5TEXT.

```
OP.MODEL CEQU OP.machine
```

To change the default organization for actual key (AK), direct access (DA), or indexed (IS) files from version 2 (ORG=NEW) to version 1 (ORG=OLD), change the DEF CB5\$xxOLDNEW statement in ASSEMOP to read the following. Only the routine PROCTAB need be compiled.

```
DEF CB5$xxOLDNEW ≡≠OLD≠≡; (xx is AK, DA, or IS)
```

## COMPILER PROGRAM LIBRARY STRUCTURE

Because the compiler was written in two languages (SYMPL and COMPASS), the order of programs on the compiler program library differs from the order in which the programs are loaded.

The PL is divided into a number of sections, by type of deck and overlay. The common decks are first, the texts second, and so on. Within each section, the decks are in alphabetic order. Common decks that call other common decks follow the common decks that are called.

End of records provided by decks named CWEORn separate the texts, COMPASS compiler code, SYMPL compiler code, the compiler skeleton, and the object routine sections.

The compiler skeleton contained in the deck -SKEL- manages the order of loading the routines. A COPYLM is performed against the assembled -SKEL- deck using all of the compiler binaries. This results in a binary file in the correct load order.

A full Update of the program library writes the texts, the COMPASS compiler code, the SYMPL compiler code, the skeleton, and the COMPASS object routine code to the compile file. The texts are used to assemble the COMPASS and SYMPL code, and CB5TEXT is used later to supply error messages (via the PP routine D00) at object time. The SYMPL compiler code calls the appropriate common decks to obtain installation parameter definitions.

When all has been assembled, the COMPASS and SYMPL compiler code is run against -SKEL- using COPYLM to produce the following overlay structure.

- 0,0 COBOL5 and other control routines. Also in this overlay are CDC CYBER Record Manager, Common Memory Manager, and the compiler table pointers.
- 1,0 Compiler initialization and control statement processing
- 2,1 Source statement scanning. This phase scans the source statements, processes COPY statements, and produces CTEXT for use in later phases of the compiler.
- 2,2 PICTURE analysis. Each PICTURE clause is broken up into internal information and checked for legality.
- 2,3 Data base translator. Processes CDCS Sub-schema information.
- 5,0 IDENTIFICATION DIVISION, ENVIRONMENT DIVISION, and DATA DIVISION (except for reports) lexical analysis. The CTEXT produced by 2,1 is processed into compiler internal tables.



6,0	DATA DIVISION storage analysis. Program storage is preallocated in this phase.
7,0	REPORT SECTION parsing and lexical analysis
11,0	REPORT SECTION pseudo-code generation. The GTEXT (pseudo-code) necessary to produce the specified reports is produced.
12,0 through 12,7	PROCEDURE DIVISION parsing and lexical analysis. The CTEXT for the PROCEDURE DIVISION is digested and GTEXT for each statement is produced.
14,0	Literal pooler
16,0	Cross-reference formatter
20,0	Code generation root overlay. Contains tables, pointers, and service routines common to the code generators and the assembler.
20,1	Code generation initialization, file table and data storage generation
20,2	Code generation. GTEXT produced by earlier overlays results in OTEXT input to the assembler.
20,3	Compiler assembler. OTEXT from 20,2 is turned into CDC CYBER machine instructions and the binaries are written out.
20,4	Debugging aids. This is a null overlay unless the compiler is assembled in debug mode.
20,5	Data Map formatter and terminal dump file producer.
30,0	Diagnostic formatter

### INSTALLATION INSTRUCTIONS

The compiler installation decks provide a way of placing COBOL 5 either on the regular system library or on a user library for checkout purposes.

The first job, PL60I, does a full Update and assembly of the compiler and object routines. It produces a new tape with the same structure as the release tape, and optionally, either creates or updates user libraries.

Since a full assembly of the compiler is a lengthy process (up to 2 hours clock time on a CDC CYBER 70 Model 73), a second job, PL60II, is provided which, using the most recent PL60 output tape, does an UPDATE,N. and assembles only the routines modified. It then executes COPYLM against the existing binaries to produce an updated compiler. This is useful if there are problems with only a few routines or if the CDCS interface requires changing. This job produces a tape and libraries the same as PL60I, provided that none of the CRM and CMM interfaces have changed and that all decks are properly ordered on the compile file.

A third job, PL60E, replaces the compiler and object routines on the running system or user libraries with those on the tape created by either PL60I or PL60II. The verification program, PL60V, can then be run to assure correct installation.

The most efficient method for producing upgraded COBOL 5 binaries, assuming that no code modifications have been made or are necessary since the last full assembly, is to execute the PL60O job. This job produces a new tape for which the binaries can be introduced into the running system or user libraries through PL60E. PL60O can be used to reformat the overlays by using relocatable binaries from the last PL60I output tape.

The use of the LOCLIB parameter in the installation decks (refer to part I, section 1) provides users with a much more flexible approach to compiler maintenance. Users might choose, for example, to use job PL60I to apply PSR code from a PSR mini-tape against their current compiler tape, creating a user library as well as a new tape. They now have a user library which can be easily tested (a memo to their users can usually supply that), and a tape that matches the library. After users are satisfied that the new compiler has no regressions, they can run job PL60E to EDITLIB the new version onto the system.

Running the same job (PL60E) with a =DEFINE LOCLIB card and using the old PL tape produces a user library for backup. When the new compiler is running trouble-free, the old compiler user library can be purged.

If the compiler is run from a file (not a user library), it produces binaries with LDSET(LIB=COB5LIB) directives in them. These may produce nonfatal LOADER diagnostics if the object routines are not present on a library with that name.

The compiler installation job decks PL60I, PL60II, PL60O, and PL60E, as well as the verification program PL60V, can be obtained from the Installation Decks PL using the procedures outlined in part I, section 1.

The following compiler routines need to be assembled when either the CDCS feature or the CMU option is to be turned on.

To activate CDCS, assemble

CGENTXT, DAIO, ETABLES, GBRANCH, GIO, NEXTR, SFETS, TABLES in COMPASS

All SYMPL texts

All SYMPL routines referenceing the common deck ASSEMOP

All SYMPL routines in the 2,1 and 2,3 overlays

To activate CMU, assemble

CGENTXT, GANMOVE, GCMUMOV, GCONDIT, GETIPS, GMOVAN, GMOVLIT, GMOVSA, COBTIME,

GMOVSUB, NEXTR, PUTPRFX, SFETS, TABLES in COMPASS

Nothing in SYMPL

All object routines should be reassembled for both options.



## RELEASE MATERIALS

The COBOL 4/5 Conversion Aid is released on tape PL69. The structure of the release format PL69 tape is as follows.

File 1	COBOL 4/5 Conversion Aid source in UPDATE program library format
File 2	COBOL 4/5 Conversion Aid absolute binary
File 3	COBOL 4/5 copy utility absolute binary
File 4	COBOL 4/5 Conversion Aid binary syntax file

## HARDWARE CONFIGURATION

The COBOL 4/5 Conversion Aid requires the same minimum hardware configuration as the NOS/BE, except that execution field length may exceed that available on a 49K CM computer (refer to the following discussion under Installation Parameters).

## GENERAL DESCRIPTION

The COBOL 4/5 Conversion Aid is a language conversion system to assist in converting CDC CYBER COBOL 4 source programs to CDC CYBER COBOL 5 source programs. Usage instructions are published in the COBOL 4/5 Conversion Aid Reference Manual.

## INSTALLATION PROCEDURE

Jobs PL69I, PL69C, and PL69V can be obtained from the installation deck program library using the procedure outlined in part I, section I of this document. These jobs should be examined and modified to accommodate permanent file ID values as well as COBOL 4/5 conversion aid installation parameters.

The installation jobs function as follows.

- PL69I updates the program library and produces a new program library tape including the three execution time binary files. If CATALOG is defined during the extraction of PL69I, the three execution files are saved as permanent files from which they may be executed.
- PL69C copies the three COBOL 4/5 Conversion Aid binaries from either the release tape or the tape written by PL69I, and catalogs them as permanent files from which they can be executed.
- PL69V is a verification job which can be used to validate proper creation of the COBOL 4/5 conversion aid permanent files. The PL69V job uses the permanent file cataloged by either PL69C or PL69I with CATALOG defined.

## INSTALLATION PARAMETERS

Installation variables for the COBOL 4/5 conversion aid can be activated during the PL69I update of the program library by the following directives.

*DEFINE CBLCOPY	Causes generation of the conversion aid version capable of handling COBOL 4 source programs containing COPY (from library) statements.
*DEFINE COPLST	Causes generation of a language conversion system in which COPY statements encountered in the COBOL 4 source program are retained as real COPY statements in the COBOL 5 source. If COPLST is not defined, the COBOL 4 COPY statements are retained only as comments, and the COPY source statements, having been made available by the CBLCOP process, are inserted in-line in the COBOL 5 source program.
*DEFINE LTAB	Refer to the following discussion.
*DEFINE LTAB,XLTAB	Refer to the following discussion.

The COBOL 4/5 conversion aid may overflow some tables while converting programs with large numbers of symbols or lengthy statements. Each name table entry is a variable length of  $4 + (n+9) / 10$  CM words for each user-defined source program name of n characters. The COBOL 4/5 conversion aid can be reinstalled, enlarging the tables, by running job PL69I (defining LTAB or LTAB,XLTAB) and cataloging the resultant files.

Name table size relates to execution FL as follows.

	<u>Default</u>	<u>*DF LTAB</u>	<u>*DF LTAB,XLTAB</u>
Name table length	3200D	6500D	14000D
Execution FL	71000B	102000B	122000B

If CBLCOPY is defined, add 12000B to the preceding execution field length requirements.

Default installation is with none of the preceding symbols defined.

RELEASE MATERIALS

CDC CYBER Control Language (CCL) is released on tape PL70. The format of the PL70 tape is as follows.

File 1 CCL source code in UPDATE program library format  
 File 2 CCL absolute binary

HARDWARE CONFIGURATION

CCL requires the minimum NOS/BE configuration.

GENERAL DESCRIPTION

CCL allows a job to conditionally skip or repeat control statements and to process control statements obtained from a separate file or from a library. CCL consists of three absolute overlays with entry points and verb table entries for each CCL verb as follows.

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

NOTES AND CAUTIONS

If a CCL verb must be changed because of a conflict with an existing library-resident program, both the entry point name and the verb table entry must be changed in the associated deck.

The verbs IFE and IF are synonymous; either or both may be defined.

INSTALLATION PARAMETERS

All installation parameters are defined in deck CCL on the UPDATE program library. The Maximum Value column shows the largest value the parameter may have.

<u>Parameter</u>	<u>Released Value</u>	<u>Maximum Value</u>	<u>Description</u>
IP.FPC	10	10	Maximum number of characters in a formal parameter
IP.SCS	40	80	Maximum number of characters in a parameter value specification
IP.LCS	10	10	Maximum number of characters in a label character string
IP.PNL	50	1023	Procedure nesting limit
IP.FP	50	500	Maximum number of formal parameters
IP.DPF	1		Specifies one of the following. 1 The default procedure file specified with IP.DPFN is defined. 0 No default procedure file name is defined.
IP.DPFN	PROCFIL		Default procedure file name used if not specified on the BEGIN statement.

<u>Parameter</u>	<u>Released Value</u>	<u>Maximum Value</u>	<u>Description</u>
IP.ATT	1		Specifies one of the following.  1 An automatic attach is done. If the procedure file is not local to the job, CCL attempts to attach a permanent file with the same name using the ID specified with IP.ID.  0 No automatic attach is done.
IP.ID	PUBLIC		Indicates the ID if automatic attach is specified (IP.ATT is set to 1).
IP.SYS	4		Defines the value of the symbolic name SYS in CCL expressions.
IP.VER	446		Defines the value of the symbolic name VER in CCL expressions.
IP.EXP	100	100	Maximum number of operands and operators allowed in a CCL expression. For each unit this parameter is decreased, the execution size of CCL is reduced by two words.

#### INSTALLATION PROCEDURE

Obtain job decks PL70I and PL70E from the Installation Decks PL tape, as described in part I, section 1. PL70I is a maintenance job deck that can be used to generate a new PL70 tape containing a revised program library and absolute binary. PL70E enters CCL into the running system from either the released PL70 tape or the new PL70 tape generated by PL70I.

The CDC CYBER 171 7152 Mass Storage/Magnetic Tape Controller installation is a two-step procedure. First, build a deadstart tape configured only for INTERCOM. Then use INTERCOM to initiate the installation decks.

## REQUIRED MATERIALS

The following materials are required to install NOS/BE on the no-card-reader minimum configuration.

- CDC CYBER Control Language (CCL) procedure file named AUTO (on the unconfigured deadstart tape)
- CCL procedure file named MUXCR
- INTERCOM 4 and CCP binaries or INTERCOM 5 and CCI binaries
- Changes to the existing DST3 deck with code enabled by =DEFINE CYB171 and =DEFINE INT5
- Installation deck named MUXCRE which installs the MUXCR procedure

## RESTRICTIONS

The following restrictions are imposed.

- The AUTO procedure can extract only decks required for installing INTERCOM and the 2550 controlware.
- Any CDC CYBER 171 installation options are defined by the AUTO procedure and cannot be modified by the site.
- The maximum number of ports allowed for the 2550 Multiplexer is 16; the first port is defined as empty.
- No site or station addresses can be specified for a port entry.
- Only MODE 4 and ASYNC ports can be specified for INTERCOM 5.

## REQUIRED MACROS

The following macros are required throughout the installation procedure.

### AUTO MACRO

The format of the AUTO macro is as follows.

AUTO, P=deck, D=density, V=vsn, PW=password, INT=version.

deck	Name of the installation deck to be loaded. Options are PL12E, PL61E, DST3, or MUXCRE. Default requests the BCC tape and catalogs the installation decks subset.
density	Density of the tape being read. Default is HY (800 bpi).
vsn	Volume serial number of the BCC tape; applicable only to BCC tape. Default is BCCTAP.
password	Initial INTERCOM unrestricted password.
version	Version of INTERCOM.
	4 INTERCOM 4 (default)
	5 INTERCOM 5

## MUXCR MACRO

The format of the MUXCR macro is as follows.

MUXCR, NAME=symbol, TM=port, SP=speed, HW=flag, RN=count, PORTS=address, ESTO=est, CH=channel, EQP=eqp

symbol	COMPASS symbol associated with the MUX and EST macros. The first character must be alphabetic.
port	Character string appended to characters TM for port specification. Options are 3, 4, A4A, and B4A. Default is 0, which specifies an empty port.
speed	Line speed for port specification. Options are any valid line speed such as 110, 300, 600, 1200, and so on. The MUXCR macro specifies the default value.
flag	Flag to specify whether or not port is hard-wired. If nonzero, the port is hard-wired. Default is 0, which specifies a dial-up port.
count	Repeat count for the number of identical port entries to be specified. This value must not exceed PORTS minus 1. Default is 1.
address	Largest decimal port address which can be specified for the MUX macro. The first site address is always an empty port; therefore, only address minus 1 ports can be specified. Also, address must be greater than or equal to 2 and less than or equal to 16. Default is 2, which allows a subsequent call to specify one user port.
est	Equipment status table number for the EST macro. Default is 1.
channel	Hardware channel to which the multiplexer is connected for the EST macro. Default is 0.
eqp	Equipment number of the multiplexer for the EST macro. Default is 7.

## INSTALLATION PROCEDURES FOR INTERCOM 4

The following procedure is used to install the CDC CYBER 171.

1. Deadstart the 66x using the coldstart procedure for 66x tape controllers as described in the NOS/BE Operator's Guide.
2. Deadstart the 66x using the unconfigured deadstart tape.
3. Create an intermediate deadstart tape with INTERCOM capability as follows.

- a. Type the following at a clear control point n.

n. X AUTO.

This generates the following request.

REQUEST (OLDPL, HY, NORING, VSN=BCCTAP)

The installation decks necessary for creating a deadstart tape are cataloged as a permanent file after the tape is assigned.

- b. Type the following at a clear control point n.

n. X AUTO, P=PL12E.

This causes installation deck PL12E to be processed, which EDITLIBs INTERCOM into the running system.

- c. Type the following at a clear control point n.

n. X AUTO, P=PL61E.

This causes installation deck PL61E to be processed, which EDITLIBs CCP routines into the running system.

- d. Type the following at a clear control point n.

n. X AUTO, P=MUXCRE.

This causes installation deck MUXCRE to be processed, which EDITLIBs the MUXCR procedure file into the running system.

- e. Type the following at a clear control point n after MUXCR has completed.

n. X MUXCR, NAME=MUX1, PORTS=6, ESTO=3, CH=1, EQP=5.

This creates a file with the following contents.

```
*IDENT MUXCR
*I EST.1
FE EST CH=1, EQP=5, ESTO=3, MUX=MUX1-T.ITABL
*C CMR
*I MUX.1
MUX1 MUX2550 6
      EMPTY
```

- f. Define ports by typing the following example at clear control points n.

**NOTES**

The NAME=symbol keyword cannot be specified during this step. If it is specified, the file is overwritten, and step e must be performed again.

```
n. X MUXCR, TM=A4A, SP=9600, HW=1.
n. X MUXCR, RN=3.
n. X MUXCR, TM=3.
```

If the RN keyword is not specified, RN=1 is assumed; therefore, the total repeat count for the preceding calls is 5. This is one less than the 6 specified by the PORTS= address keyword from step e.

The file created in step e has the following contents.

```
*IDENT MUXCR
*I EST.1
FE EST CH=1, EQP=5, ESTO=5, MUX=MUX1-T.ITABL
*C CMR
*I MUX.1
MUX1 MUX2550 6
      EMPTY
      TMA4A 9600,,HW
      EMPTY
      EMPTY
      EMPTY
      TM3 *
```

- g. Type the following at a clear control point n.

n. X AUTO,DST3,PE.

This causes the installation deck DST3 to be processed, which assembles the new central memory resident, captures the running system, and writes a new deadstart tape at PE density (1600 cpi).

4. Deadstart (warmstart) using the new deadstart tape.  
5. Type the following at a clear control point n prior to bringing up INTERCOM.

n. X AUTO,PW=INSTALL.

This creates a password file which allows the user to log in as follows.

```
LOGIN,userid,INSTALL
```

More than one user can log in, but each user must specify a unique user identifier (userid).

6. Install NOS/BE using INTERCOM.

## INSTALLATION PROCEDURES FOR INTERCOM 5

The following procedure is used to install the CDC CYBER 171.

1. Deadstart the 66x using the coldstart procedure for 66x tape controllers as described in the NOS/BE Operator's Guide.
2. Deadstart the 66x using the unconfigured deadstart tape.
3. Create an intermediate deadstart tape with INTERCOM capability as follows.

- a. Type the following at a clear control point n.

n. X AUTO, INT=5.

This generates the following request.

```
REQUEST (OLDPL, HY, NORING, VSN=BCCTAP)
```

The installation decks necessary for creating a deadstart tape are cataloged as a permanent file after the tape is assigned.

- b. Type the following at a clear control point n.

n. X AUTO, P=PL14E.

This causes installation deck PL14E to be processed, which EDITLIBs INTERCOM into the running system.

- c. Type the following at a clear control point n.

n. X AUTO, P=PL99E.

This causes installation deck PL99E to be processed, which EDITLIBs CCI routines into the running system.

- d. Type the following at a clear control point n.

n. X AUTO, P=MUXCRE.

This causes installation deck MUXCRE to be processed, which EDITLIBs the MUXCR procedure file into the running system.

- e. Type the following at a clear control point n after MUXCR has completed.

n. X MUXCR, NAME=MUX1, PORTS=6, ESTO=3, CH=1, EQP=5.

This creates a file with the following contents.

```
*IDENT MUXCR
*I EST.1
FE EST CH=1, EQP=5, ESTO=3, MUX=MUX1-T.ITABL
*C CMR
*I MUX. 1
MUX1 MUX2550 6
      EMPTY
```

- f. Define ports by typing the following example at clear control points n.

### **NOTES**

The NAME=symbol keyword cannot be specified during this step. If it is specified, the file is overwritten, and step e must be performed again.

```
n.X MUXCR, TM=A4A, HW=1.
n.X MUXCR, TM=B4A.
n.X MUXCR, RN=2.
n.X MUXCR, TM=3.
```

If the RN keyword is not specified, RN=1 is assumed; therefore, the total repeat count for the preceding calls is 5. This is one less than the 6 specified by the PORTS= address keyword from step e.



The file created in step e has the following contents.

```
*IDENT      MUXCR
*I  EST.1
FE  EST CH=1, EQP=5, ESTO=5, MUX=MUX1-T.ITABL
*C  CMR
*I  MUX.1
MUX1      MUX2550 6
          EMPTY
          MODE 4   CL=1,MODE=4A,LT=HW,CODE=ASCII
          MODE 4   CL=0,MODE=4A,CODE=BCD
          EMPTY
          EMPTY
          ASY NC   LS=AUTO
```

g. Type the following at a clear control point n.

n. X AUTO,DST3,PE.

This causes the installation deck DST3 to be processed, which assembles the new central memory resident, captures the running system, and writes a new deadstart tape at PE density (1600 cpi).

4. Deadstart (warmstart) using the new deadstart tape.
5. Type the following at a clear control point n prior to bringing up INTERCOM.

n. X AUTO,PW=INSTALL.

This creates a password file which allows the user to log in as follows.

LOGIN,userid,INSTALL

More than one user can log in, but each user must specify a unique user identifier (userid).

6. Install NOS/BE using INTERCOM.



RELEASE DESCRIPTION

INTERCOM 5 in conjunction with the NOS/BE operating system provides TTY and CRT terminals with time-shared access to CDC CYBER 170, CDC CYBER 70, and 6000 Series computers. Also, remote batch jobs can be submitted from terminals equipped with a remote card reader and printer, or from a low- or medium-speed batch terminal. Programs written in the FORTRAN, COBOL, ALGOL, COMPASS, or BASIC languages can be submitted from a remote terminal for execution at control points; the user at the remote terminal can interact with the executing program. Program output can be routed to the line printer and card punch at the central site or to a terminal equipped with line printers or card punches. Through the system permanent file feature, input from a central site magnetic tape or card reader is available to the remote user.

HARDWARE CONFIGURATION

In addition to the minimum hardware required by the NOS/BE system, INTERCOM 5 requires the following equipment for communication and operation.

- A CRT terminal, model 214-11, 214-12, 217-11, 217-12, 217-13, 217-14, 711-10, 731-12, 732-12, 734 Remote Batch Terminal, 714-10, 714-20, or CDC CYBER 18 or a model 33, 35, or 38 KSR or ASR Teletype terminal, or a 713 TTY-compatible terminal, or a 751 TTY-compatible terminal, or a HASP workstation, IBM 2780, or IBM 3780 batch terminal.
- A dedicated 255x Network Processing Unit (NPU) on a dedicated channel for TTY and/or CRT terminals.
- Data Sets for communication between the remote terminal and central site. Teletype terminals require 103A or 212A Data Sets; CRT terminals require 201A, 201B, or 201C Data Sets, or CDC 358 Transceivers. Refer to the Control Data Communications Handbook for specific details of the exact modem strapping option required by INTERCOM 5.

REQUIRED HARDWARE OPTIONS

711-10	
Data control	711-102
714-10 or 714-20	
Display (8 x 80 or 16 x 80)	714-122 or 714-123
731-12	
732-12	
Memory increment (8K bytes)	730-100
Display (16 x 80)	730-101

HARDWARE OPTIONS

Teletype	
Paper tape reader/punch	
217-11, 217-12, 217-13, 217-14	
Card reader	224-11, 12, 13, or 14
Line printer	222-11, 12, 13, or 14
711-10	
Memory option (16 x 80 screen)	711-100
Character printer	711-120 or 711-21
714-10	
Display (8 x 80 or 16 x 80) (up to 8 additional)	714-122 or 714-123
Character printer (up to 3)	711-120 or 711-121

## RELEASE MATERIALS

INTERCOM version 5 release material consists of a magnetic tape (PL14) containing the INTERCOM program library as file one.

## NOTES AND CAUTIONS

Some mode 4 CRT terminals work properly in all respects except that they do not properly process the sequence bit; generally, they send a zero sequence bit in all transmissions to the 6000 computer. This problem is a terminal malfunction, a loose wire, or bad hardware card. Such terminals may undergo endless retransmissions of one of the first two WRITES to the CRT screen. Should this retransmission be observed regularly when a particular terminal connects, hardware support personnel should check the sequence bit (bit 24) in the station address word.

### LINE SKIPPING ON 714 NONIMPACT PRINTER

When column 80 is reached, an INTERCOM-generated line skip occurs. 714 nonimpact printers have a photo cell switch set at column 80 to skip a line. Thus, when input lines exceed 79 characters, output is double-spaced. If this is not desirable, request the site customer engineer to move the photo cell switch beyond column 80.

### ERROR PROCESSING ON 711/714

On the 714, the ERR enable/disable switch should be set to ENABLE. On the 711, the ERR disable cable should not be present. This allows the controller to give an ERR response both to writes to nonexistent stations, and to invalid messages in general.

### CHARACTER SET SUPPORT

The IP.CSET display code character set selection affects INTERCOM only with respect to 63- or 64-character set selection. The BCD and ASCII printer character set default and 026/029 keypunch code default selections affect only the remote batch terminals. How it affects each one is described in each terminal reference manual.

In addition to the IP.CSET display code character set selected, INTERCOM allows users to select extended ASCII 95- or 256-character sets for communication with a mode 3 type terminal. These sets are described in the INTERCOM Reference Manual.

### LIMITATIONS AND SYSTEM CONSIDERATIONS

When the CONNECT command (or CONNEC call) is used, the specified data is routed to or from the terminal each time the file is read or written. When simultaneous operations are to be performed, no more than one file should be connected to a terminal for interactive operations at any time.

The PASSWRD utility (refer to Password File Creation) should not be run while INTERCOM is up because numerous problems may occur if a user id is changed while INTERCOM is up. These include remote batch file security problems and EDITOR's edit files being swapped between users.

### GENERAL PROCEDURES

Installation of a complete INTERCOM system requires establishing installation parameters and installing from the INTERCOM OLDPL. The card deck described later can be run at the central site to install INTERCOM. FORTRAN Extended 4 and COMPASS must be installed before INTERCOM can be installed.

## INSTALLATION PARAMETERS

The following items should be considered when configuring the INTERCOM system for a particular installation.

- Parameters in the INTERCOM common deck INTCOM can be changed to affect the characteristics of INTERCOM.
- An equipment status table (EST) entry must be established for each multiplexer dedicated to INTERCOM.
- In CMR, a multiplexer table must be defined which contains subtables for each multiplexer dedicated to INTERCOM.
- Certain tables within 1CI, 1QP, and 3TT can be set to control use of selected commands.
- Parameters in the EDITOR common decks IPFTN and IPCOM can be changed to affect the characteristics of EDITOR.
- Parameters in the multiuser job common decks MUJCOM and CMUJCOM can be changed to affect the characteristics of multiuser jobs (particularly EDITOR).

## INTERCOM COMMON DECK SETTINGS

Release values are shown in the following list of INTERCOM parameters for the common deck INTCOM present on PL14. If these parameters are to be changed, the cards containing the proper code with the CEQU macro should be placed after an \*INSERT INTCOM.43 directive and inserted into the first update record of the deck PL14I. Alternate tested values are shown in parentheses.

A cross-reference listing showing the routines that reference each INTCOM symbol appears in part III of this document.

IP.FTNTS      CEQU      0

This parameter specifies the installation default FORTRAN Extended 4 compiler for EDITOR. A value of zero specifies OPT=0; a value of 1 specifies Time-Sharing.

IP.IACES      CEQU      11

An 11-bit field contains the user table access field and user permission bits. This value must be the same as the value for IP.IACES (refer to NOS/BE IPARAMS in this section). The entire 11-bit field is used to determine if a user has access to a specific utility or routine. The setting of IP.IACES determines how many bits, right-justified, are to be used as the access level. The remaining bits (11-IP.IACES) are used as permission bits.

User access level is an octal integer (range 0 to (2\*\* IP.IACES)-1) and is contained in the user table after the user logs in. User's access level must be greater than or equal to the access level of the command in order to use a command.

Permission bits form a mask constant (range 0 to (2\*\*(11-IP.IACES))-1). Each bit which is set in the command permission-bit mask must also be set in the user's permission-bit mask in order for the user to use the command.

1AJ and LOADER check permission bits and access levels for commands found in the NUCLEUS Entry Point Name Table.

A program in a library, specifically the Entry Point Name Table entry in the NUCLEUS library, has an 11-bit permission bits/access level value. In addition, only this type of command verb has one additional bit associated with it indicating whether the entry is control-statement-callable. In the EPNT entry, bits 14-4 contain the permission bits and access level required; bit 3 contains the control-statement-callable bit (0 = not control statement-callable). 1AJ checks bit 3 for all control statements.

EDITLIB allows definition of permission bits and access levels via the SETAL directive or the AL parameter of the ADD and REPLACE directives. This value is not access level; it is a 12-bit value combining permission bits (upper 11-IP.IACES bits), access level (bits IP.IACES-1), and control-statement-callable (bit 0). The upper 11 bits of this value are the required permissions and access level found in bits 14-4 of the EPNT entry.

During a PASSWRD run, a user's permissions and access level are defined via the A=accevl parameter. This value is an 11-bit octal number combining permission bits and access level. No control statement-callable value is associated with the user's accevl value.

IP.IACES may be given any value between 0 and 11. If IP.IACES = 0, then the entire field is permission bits. If IP.IACES = 11, then the entire field is access level.

Example

1. IP.IACES = 6
2. EDITLIB run with directives  
 SETAL (FILES, 201)  
 SETAL (ASSETS, 407)
3. PASSWRD run with directives  
 ADD U=USER1, P=PASS1, A=2  
 ADD U=USER2, P=PASS2, A=302  
 ADD U=USER3, P=PASS3, A=3077  
 ADD U=USER4, P=PASS4, A=1515  
 ADD U=USER5, P=PASS5, A=0712
4. As the result of the preceding installation, the following relationships exist.

<u>COMMAND</u>	<u>PERMISSIONS ASK</u>	<u>ACCESS LEVEL</u>	<u>PERMISSIONS REQUIRED</u>
FILES	1	0	0
ASSETS	2	3	1

<u>USER</u>	<u>PERMISSIONS ASK</u>	<u>ACCESS LEVEL</u>	<u>PERMISSIONS GRANTED</u>
USER1	0	2	NONE
USER2	3	2	0,1
USER3	30B	77B	3,4
USER4	15B	15B	0,2,3
USER5	7	12B	0,1,2

	<u>FILES</u>	<u>ASSETS</u>
USER1	P0	P1, AL
USER2	ALLOWED	AL
USER3	P0	P1
USER4	ALLOWED	P1
USER5	ALLOWED	ALLOWED

USER X COMMANDS ALLOWED

- Pn Denied because user lacks permission n
- AL Denied because user access level too low

IP.ID CEQU 1

If one, the INTERCOM user id is used as the default permanent file id by commands STORE, FETCH, and DISCARD. If zero, the permanent file id must be specified by the INTERCOM user.

IP.IDFL CEQU 55000B

Default field length assigned to a user's program when the user has not entered a field length (EFL).

IP.IDFLE CEQU 0

Default ECS field length (in multiples of 1000g) allowed a user if no E parameter was specified for the user's id on the password file.

IP.IHEAD CMICRO 0, (CONTROL DATA INTERCOM 5.0)

Header output by 1IM when a remote terminal dials into the INTERCOM system.

IP.IIBMN CEQU 12

Minimum number of interactive empty buffers needed; these buffers are maintained by INTERCOM.

IP.IIBMX CEQU 18

Maximum number of interactive empty buffers needed.

IP.IM3BS CEQU 4095

Default page size in characters for mode 3 terminals. This value should be  $20 \leq x \leq 4095$ .

IP.IM3LW CEQU 72

Default line length in characters for mode 3 terminals. This value should be  $10 \leq x \leq 132$ .

IP.IPRLS CEQU 100B

Priority loss per 100g PRUs used by an output file after the first 100g PRUs (refer to IP.MPRIT description).

IP.I4ABS CEQU 1040

Default screen size in characters for mode 4A terminals. This value should be  $20 \leq x \leq 1920$ .

IP.I4ALW CEQU 80

Default line length in characters for mode 4A terminals. This value should be  $10 \leq x \leq 80$ .

IP.I4CBS CEQU 1280

Default screen size in characters for mode 4C terminals. This value should be  $20 \leq x \leq 1920$ .

IP.I4CLW CEQU 80

Default line length in characters for mode 4C terminals. This value should be  $10 \leq x \leq 1920$ .

IP.I27BS CEQU 400

Default block size in characters for IBM 2780 terminals. The value should be  $20 \leq x \leq 800$ .

IP.I27LW CEQU 80

Default line length in characters for IBM 2780 terminals. The value should be  $10 \leq x \leq 80$ .

IP.I37BS CEQU 400

Default block size in characters for IBM 3780 terminals. The value should be  $20 \leq x \leq 800$ .

IP.I37LW CEQU 80

Default line length in characters for IBM 3780 terminals. The value should be  $10 \leq x \leq 80$ .

IP.IHSBS CEQU 400

Default block size in characters for HASP terminals. The value should be  $20 \leq x \leq 800$ .

IP.IHSLW CEQU 80

Default line length in characters for HASP terminals. The value should be  $10 \leq x \leq 80$ .

IP.BSIM CEQU 1 (0)

Default bisynchronous input mode for card reader files on HASP, IBM 2780 and IBM 3780 terminals; 1 selects 026 punch codes and 0 selects 029 punch codes.

IP.IND CEQU 1 (2)

Maximum number of active INTERCOM drivers (of any type) allowed in the system simultaneously. It should never exceed six.

IP.ISFL CEQU 2500B

Default swap-in field length for INTERCOM. The swap-in field length is the amount of memory requested to swap in an INTERCOM command.

IP.MALOC CEQU 4000B

A 12-bit octal value defining the allocation style for files created by a multiuser job. Bit 11 always is set to one to indicate that a permanent file device is requested. Bits 5 through 0 indicate the allocation style. This value is placed in the File Name Table entry generated for new multiuser job files, in byte C.FALLOC.

IP.MPRIT CEQU 4000B

Maximum priority to be assigned to an output file diverted by INTERCOM. If  $f_1$  is the length of the file in PRUs, the priority assigned to a file can be expressed as  $IPMPRIT - (IP.PRLS * (f_1 - 100_g) / 100_g)$ , where / denotes an integer divide.

IP.MXCOR CEQU 2500B

Maximum field length allowed for INTERCOM buffer usage (in multiples of 100 octal words); cannot exceed 4000g.

IP.PRIX CEQU 3777B (7000B)

Nonzero indicates the priority given to input files read from remote site. If zero, priority will be taken from Job card.

IP.TSL CEQU 10B

Default time limit in seconds for execution of a user's program, if the user has not entered a time limit (ETL).

IP.BUFFE CEQU 10B

Number of PRU (physical record unit) buffers allocated for each 255x (FE) entry in the multiplexer subtable. These buffers are used by 1ND to transfer remote batch data to and from disk. The buffers are referred to as PRUBs.

IP.PRUB CEQU 2

Number of PRUs allocated to each PRUB

IP.X780 CEQU 0

Specifies the default for automatic terminal detection for the BISYNC macro. If zero, the IBM 2780 is selected; if 1, the IBM 3780 is selected.

LE.IPHDR CEQU 9

Length of the header for a PRUB. The buffer header contains the FET information and is also used by the 1ND driver to store usage statistics.

The word length of a PRUB can be determined by using the preceding three parameters in the following formula.

$$\text{Length} = \text{IP.BUFFE} * ((\text{IP.PRUB} * 64) + \text{LE.IPHDR} + 1)$$

Q.ILNOFC CEQU 1130B

This value is a timer for 1ND. It is used to turn an OFF line back ON. The release value is about 10 minutes.

#### NOS/BE IPARAMS SETTINGS

These parameters must be set at \*INSERT IPARAMS.15 when NOS/BE is installed (deck PL1A1).

IP.IACES CEQU 11

Defines the number of bits in the access level, for use by 1AJ and LOADER. This value must be the same as that specified for the INTERCOM parameter IP.IACES.

IP.ILCMD CEQU 1

If set to 1, the last word in the user table will store the last command entered by each user for display on the DSD Q display. If 0, it will not be used for this purpose.

IP.IUSID CEQU 2RAJ

Defines the first user id available for assignment by the program PASSWRD. The value of this parameter is determined by the number of hardwired remote batch terminals defined in the system. The hardwired remote batch terminals use one id per terminal.

This user id is the lowest available to be assigned an interactive user. Every remote high speed batch terminal connected to the system must have its own terminal id assigned to it.

A cross mapping of referencing routines and all symbols in IPARAMS (IPTEXT) can be found in part III.

#### NOS/BE PPSYS SETTINGS

The symbol IP.INT5 must be defined at ident PPSYS.11 when NOS/BE is installed.

```
*D F7220PP.6
IP.INT5 EQU 1
```

Setting IP.INT5 to 1 allows INTERCOM 5 to be assembled properly. The default value is 0, which causes NOS/BE to be assembled for INTERCOM 4.



## EST ENTRY

The EST table, established when deck PL1A1 is run to install NOS/BE, must contain an entry for each multiplexer dedicated to INTERCOM. The channel referenced in this entry must be dedicated to the INTERCOM multiplexers on that channel. For nonallocatable equipment, the EST uses the EST macro which has been modified as follows.

type EST parameters (of the form key=value)

Macro parameters used by INTERCOM include the following.

type FE for 255x  
CH= Channel for multiplexer or 7077 Communication Station or 255x  
EQP= Equipment number for 255x Front End  
MOD= OFF if off; otherwise do not use  
MUX= Index to INTERCOM multiplexer table

A typical EST entry might appear as follows.

```
*I EST . 1  
FE EST CH=3, EQP=5, MUX=MUX1-T.ITABL
```

This entry notifies the multiplexer driver that a 2550 with equipment number 5 is on channel 3; the index to the multiplexer subtable for this 2550 is MUX1; and T.ITABL is the beginning of the multiplexer table.

## CONFIGURATION PARAMETERS (INTERNAL TO CMR)

This parameter defines the length of the INTERCOM multiplexer table. It must be set at \*INSERT CMRIP.1 when NOS/BE is installed. The default value follows.

```
L.ITABL CEQU 19
```

This parameter should be changed to reflect the size of the multiplexer table for each installation. The length of the table can be determined from the following formula.

```
L.ITABL=2+N50+N50PORTS  
N50          Number of 255xs.  
N50PORTS    Total number of 255x ports defined.
```

## CMR MULTIPLEXER TABLE

The CM resident INTERCOM multiplexer table is used by INTERCOM to provide data on the hardware configuration of the installation and to record parameters. It consists of two dedicated parameter words and one or more subtables assigned to the multiplexers serviced by INTERCOM.

The first two words of the multiplexer table, the parameter words, start at location T.ITABL in CMR. They are already assembled into CMR. The subtables follow the parameter words in any order convenient to the installation. The first subtable must be defined at \*INSERT MUX.1 when NOS/BE is installed. Each subtable has a relative pointer in the EST entry for that multiplexer. The upper bound of the multiplexer subtable must not extend beyond 7777B.

For 255xs, port 0 must be empty and baud rates must be specified in descending order for ports.

## CMR MULTIPLEXER SUBTABLE GENERAL FORMAT

Each multiplexer subtable contains one macro to define the type of multiplexer, followed by one macro for each port defined on that multiplexer. The address of the macro describing the multiplexer is the same address used in the EST entry defining that multiplexer. A subtable for a multiplexer might be defined as follows.

```
MUX 1  MUX2550  4  
EMPTY  
MODE 4  MODE=4A, CARR=CONST, LT=HW  
BISYNC  MODE=HASP  
ASYNC   LT=HW, LS=300
```

ASYNC LS=AUTO

When a 255x Front End is configured, it is advisable to place the highest speed terminals on the lowest ports and to place any empty ports at the high number port positions. Thus, the 255x should be configured with 9600 baud terminals first, then 4800 baud terminals, then 2400 and 2000 baud terminals, then TTYs, then empty ports.

#### MULTIPLEXER DEFINITION ENTRIES

INTERCOM 5 recognizes only a 255x communication subsystem. It is defined with the following macro.

MUX2550 Number of ports (maximum 127)

The number of ports parameter indicates the highest number port+1 which INTERCOM is to service on that multiplexer.

#### PORT DEFINITION ENTRIES

Currently, four types of ports are recognized by the CMR macros for a 255x. They are defined with the following macros.

ASYNC Any mode 3 terminal (TTY 33/35/37/38, 713)  
MODE4 Any mode 4 terminal (200UT, 711, 714, 214, 217)  
BISYNC IBM 2780, IBM 3780, or HASP terminal  
EMPTY Empty port (not serviced by INTERCOM 5)

#### **NOTE**

Port 0 must be empty.

#### MACRO DEFINITIONS

The ASYNC macro defines all asynchronous terminals and has the following format.

ASYNC keyword=xx, keyword=yy

Keywords include the following.

LT	Line type
	DU Dial-up (default)
	HW Hardwired
LS	Line speed
	110 110 baud
	150 150 baud
	300 300 baud
	600 600 baud
	1200 1200 baud
	2400 2400 baud
	4800 4800 baud
	9600 9600 baud
	AUTO Automatic baud rate recognition (110, 150, 300, and 1200 only)
CO	Carrier signal attribute
	ON Carrier initially on (default)
	OFF Carrier initially off. This value is for hardwired modems which do not have a carrier-on signal until the terminal is connected.
LO	Line ordinal. This parameter allows the installation to specify the line ordinal of the port being defined. Use of this parameter allows a site to omit EMPTY port definitions, since it causes generation of empty port definitions, if required. Line ordinals in subsequent macros must ascend in value, but need not be in sequence. Macros using the LO parameter can be mixed with macros omitting it.

The MODE4 macro defines all mode 4 synchronous terminals and has the following format.

MODE4 keyword=xx, keyword=yy,...

Keywords include the following.

MODE	Mode of terminal
4A	Mode 4A terminal (214, 217, 200UT)
4C	Mode 4C terminal (711 C/D, 714)
AUTO	4A/4C automatic terminal detection (default)
LT	Line type
DU	Dial-up (default)
HW	Hardwired
CARR	Carrier type (HW line type only)
CONTR	Controlled carrier (default)
CONST	Constant carrier
CODE	Character code of terminal (mode 4A only)
ASCII	ASCII character set (default)
BCD	BCD character set
CL=(s <sub>1</sub> ,s <sub>2</sub> ,...,s <sub>n</sub> )	Cluster address
	A list of cluster (site) addresses indicates the port is to service a multidrop line to which terminals at those site addresses can be connected. Up to 12 site addresses can be specified in any order. Omission of cluster address causes the macro to assume cluster address 0.
TA=(t <sub>1</sub> ,t <sub>2</sub> ,...,t <sub>n</sub> )	Terminal address (mode 4C, automatic terminal detection only)
	A list of terminal (station) addresses indicates the terminal has several CRT stations to be serviced. Printer stations must not be specified in the macro call. Up to nine terminal addresses, 1 through 3, 5 through 7, 9 through 11, can be specified in any order. Terminal addresses 0, 4, and 8 are reserved for printer stations. Omission of terminal address causes the macro to assume terminal address 0 on mode 4A ports and terminal address 1 on mode 4C or automatic terminal detection ports.
LO	Line ordinal. Refer to the ASYNC macro for description.

Examples of mode 4 terminal definitions follow.

MODE 4	LT=HW, CARR=CONST, MODE=4A	Defines a mode 4A, hardwired, constant carrier terminal.
MODE4	MODE=4C, LT=HW, CL=(0, 5, 2), TA=(6, 1, 2, 9)	Defines a mode 4C, hardwired port with three cluster addresses, each of which may have four terminal addresses.
MODE4	MODE=4A, CL=(0, 1, 2, 5, 6)	Defines a multidrop mode 4A party line with five possible cluster (site) addresses.

The BISYNC macro defines a bi-synchronous terminal (IBM 2780, IBM 3780, HASP) and has the following format.

BISYNC Keyword=xx, keyword=yy,...

Keywords include the following.

MODE	Mode of terminal
2780	IBM 2780 terminal
3780	IBM 3780 terminal
HASP	HASP terminal
AUTO†	Default (IBM 2780, IBM 3780, or HASP terminal)

† Defining AUTO ensures that both the IBM 2780 or 3780 and HASP TIPS are loaded into the 255x. To save memory in the 255x, AUTO should not be used if only HASP or only IBM 2780 or 3780 terminals are to be configured.

LT	Line type
	DU Dial-up (default)
	HW Hardwired
CARR	Carrier type (HW line type only)
	CONTR Controlled carrier (default)
	CONST Constant carrier
LO	Line ordinal. Refer to the ASYNC macro for description.

Examples of the BISYNC macro include the following.

BISYNC	MODE=HASP
	Defines a HASP dial-up terminal
BISYNC	LT=DU, MODE=2780
	Defines an IBM 2780 dial-up terminal

When INTERCOM is first initiated, the INTERCOM initialization routine, I11, initiates the drivers as dictated by the multiplexers defined in the EST and the port definitions defined in the multiplexer subtables. If all equipments (multiplexers) on a channel are turned off when INTERCOM is initiated, no driver is initiated to service that channel; however, the multiplexer subtables for all of the equipment will be examined and initialized by I11.

The user should make certain that only one EST entry points to each multiplexer subtable whether the equipment is on or off.

Installation deck PL14I also will compile the relocatable multiuser job subroutines (deckname MUJSUBS). Deck PL14E will not add them to the running system for reasons of size and expected infrequency of use. However, MUJSUBS always must be included on the COMPILE file when EDITOR is compiled and loaded, so that references to the muj subroutines from EDITOR are satisfied. If a full UPDATE is done, the subroutines are included on the COMPILE file. If an UPDATE,Q is done and the EDITOR is to be modified, the UPDATE input must include a \*COMPILE MUJSUBS. (EDITOR does not use FTNMUJ or COBOMUJ, the decknames for the FORTRAN Extended and COBOL muj preprocessors.)

After the password files are established and the time has been initialized, INTERCOM should be brought up at control point zero with the console type-in INTERCOM. The INTERCOM system is then ready to service remote terminal users.

#### COMMAND TABLE STRUCTURE (ICI OVERLAY 2CS — COMMON DECK COMTBL)

Prior to INTERCOM installation, release values in the command table in 2CS can be changed or a new command or multiuser-job entry can be added. The command table is split into four parts based on the length of the command name. New entries should be inserted at the following locations (figure II-28-1).

1- or 2-character name	*1,RBS0033.9
3- or 4-character name	*1,RBS0033.18
5- or 6-character name	*1,COMTBL.58
7-character name	*1,IN40844C.14

The four command types each have an entry-definition macro as follows.

COM2CC	Defines a command processed by 2CC.
MUJ	Defines a multiuser job.
EXPCOM	Defines a remote-batch command processed by 1NP.
REMOTE	Defines a command which manipulates queue files or executing jobs.

A command-definition entry has the following general form.

name MACRO parameters

where name is the command name, such as ON, and MACRO is one of the preceding macro names.

## COM2CC MACRO

The COM2CC macro defines a command which is processed by an independent routine in overlay 2CC. The macro format is as follows.

name	COM2CC L=1, P=p, B=b, MP=mp, ADDR=ad	
l	YES	User must be logged in to use this command
	NO	User need not LOGIN if at a hardwired terminal
	Default	YES
p	YES	Command may be used while in a pause state
	NO	Command may not be used while in a pause state
	Default	NO
b	YES	Command allowed only at a batch terminal
	NO	Command allowed from any terminal type
	Default	NO
mp	Maximum number of parameters which can follow command verb; range 0-5. If MP is specified, even MP=0, parameters in the input line are counted. If the number of parameters exceeds mp, the line is rejected as a format error. Do not specify MP when commands contain parameters over 7 characters or for commands such as MESSAGE for which parameters are meaningless.	
ad	2CC address (routine name) where this command is processed. If the AD parameter is omitted, a routine with the same name as that of the command is assumed.	

## MUJ MACRO

The MUJ macro defines a multiuser job. A corresponding entry must be made in muj table of 1QP. The macro has the following format.

name	MUJ ORD1QP=ord
ord	1QP MUJ ordinal. EDITOR=1, HELLO7=2; others should proceed sequentially from 3

## EXPCOM MACRO

The EXPCOM macro defines a command processed by 1NP, and controls parameter processing for the command. The macro has the following format.

name	EXPCOM B=b, P=p, MP=mp, EXPORD=ord, PRE=pre
b	Same as for COM2CC, except default =YES
p	Same as for COM2CC
mp	Same as for COM2CC
ord	1NP command ordinal. An entry must be added at installation to the 1NP jump table for each new EXPCOM command.
pre	Address (name) of 2CC subroutine which does preprocessing (prior to extraction and validation of equipment mnemonic) for this command.

## REMOTE MACRO

The REMOTE macro defines commands which manipulate the user's queue files and execution jobs, specifically the commands DROP, KILL, DIVERT, EVICT, and PRIOR. Adding such an entry requires modifications to the 2CC routine REMOTE. Anyone contemplating this course should consult the IMS.

### MUJ TABLE STRUCTURE (1QP)

Each multiuser job as defined in the command table of 2CS also must be defined in the muj table of 1QP, MUJTABL. The position of an entry in MUJTABL is defined as the 1QP muj ordinal. Entries are made with the macro MUJTBL, at \*B 1QP.599 (figure II-28-2).

MUJTBL		name,fl,swpin,swpout,editor
name		Name of the muj
fl		Field length of muj (actual value)
swpin		Delay, in 1CI cycles (depends on IP.TICI, released for 1/2 second), between discovery of need to swap in the muj and actual entry into the scheduling queue. This value increases response time to muj requests (when the muj is swapped out) but allows requests to accumulate; so that when the muj is in, it is more likely to process multiple users. Maximum is 4095.
swpout		Delay, in 1CI cycles, between discovery of need to swap out muj and actual swap out. A high value setting essentially dedicates the muj at a control point.
editor	1	muj EDITOR
	0	otherwise

The parameters swpin, swpout, and editor may be null, and default values 1, 0, and 0, respectively, are assumed.

### TBL ASSEMBLY OPTIONS

Ten TBL command ordinals (14-23) are reserved for users to add routines to TBL. To add a routine with entry point xxx and command ordinal 14, change the fourteenth entry of TBL table TABLE to read the following.

CON    xxx

The TBL command ordinal is an index into ICPLIB. TBL tests bits 0 and 1 of table ICPLIB (12-bit entries) to determine if checks should be made for the calling program. If bit 0 is set, the calling program is a system library program. If bit 1 is set, the calling program is at an INTERCOM control point.

### TABLE CHANGES AND RELEASE SETTINGS

Changes to the tables in routines 2CS, 1QP, and TBL should be included in the UPDATE record at the directive \*/ADD CORRECTIONS HERE in installation deck PL14L. Figures II-28-1 and II-28-2 show release values and UPDATE identifiers.

VERB12	EQU	*		COMTBL	2
C	EXPCOM	P=NO, EXPORD=.CONT		COMTBL	3
E	EXPCOM	P=NO, MP=1, EXPORD=.END, PRE=END		RBS0033	4
G	EXPCOM	P=YES, B=NO, MP=1, EXPORD=.GO, PRE=GO		RBS0033	5
GO	EXPCOM	P=YES, B=NO, MP=1, EXPORD=.GO, PRE=GO		RBS0033	6
H	COM2CC	L=NO, P=NO, B=YES, MP=2		RBS0033	7
M	COM2CC	L=NO, P=YES, B=NO, ADDR=MESSAGE		IN40844K	7
ON	EXPCOM	P=NO, MP=1, EXPORD=.ON		COMTBL	9
				RBS0033	8
				COMTBL	13
VERB34	EQU	*		COMTBL	14
				COMTBL	15
BSP	EXPCOM	P=NO, MP=2, EXPORD=.BSP		RBS0033	10
CRT	COM2CC	L=NO, P=NO, B=YES, MP=1		IN5A268	4
DMP	COM2CC	L=YES, P=NO, B=NO, MP=2		COMTBL	18
DROP	REMOTE	P=YES, MP=1, REMORD=0, SUBORD=1		COMTBL	19
EFL	COM2CC	L=YES, P=NO, B=NO, MP=1		COMTBL	20
END	EXPCOM	P=NO, MP=1, EXPORD=.END, PRE=END		RBS0033	11
ETL	COM2CC	L=YES, P=NO, B=NO, MP=1		COMTBL	22
KILL	REMOTE	P=NO, MP=1, REMORD=0, SUBORD=0		COMTBL	23
LOCK	COM2CC	L=NO, P=YES, B=NO, MP=1		IN41998B	1
OFF	EXPCOM	P=NO, MP=1, EXPORD=.OFF		RBS0033	12
READ	COM2CC	L=NO, P=NO, B=YES, MP=3		RBS0033	13
REP	EXPCOM	P=NO, MP=2, EXPORD=.REPEAT		RBS0033	14
REW	EXPCOM	P=NO, MP=1, EXPORD=.REWIND		RBS0033	15
RFL	COM2CC	L=YES, P=NO, B=NO, MP=1		F7560JD	1
RTN	EXPCOM	P=NO, MP=2, EXPORD=.RETURN		RBS0033	16
SUP	EXPCOM	P=NO, MP=1, EXPORD=.SUS		RBS0033	17
TAPE	COM2CC	L=YES, P=YES, B=NO, MP=1		IN40844C	13
TERM	COM2CC	L=NO, P=YES, B=NO, MP=1		IN5A283B	1
WAIT	EXPCOM	P=NO, MP=1, EXPORD=.WAIT		RBS0033	18
				COMTBL	36
VERB56	EQU	*		COMTBL	37
				COMTBL	38
BLOCK	EXPCOM	P=YES, B=NO, MP=1, EXPORD=.BLOCK, PRE=BLOCK		RBS0033	19
CONTIN	EXPCOM	P=NO, MP=0, EXPORD=.CONT		RBS0033	20
DEFINE	EXPCOM	P=NO, MP=7, EXPORD=.DEFINE		IN5A178B	1
DIVERT	REMOTE	P=NO, MP=4, REMORD=3, SUBORD=1, QP=PARM3, MQ=ALL		COMTBL	42
EDITOR	MUJ	ORD1QP=1		COMTBL	43
EVICT	REMOTE	P=NO, MP=2, REMORD=1, SUBORD=0, QP=PARM2, MQ=ALL		COMTBL	44
HELLO7	MUJ	ORD1QP=2		S40258A	3
				COMTBL	47
LGINORD	EQU	*	FWA OF LOGIN ENTRY	IN41353	5
LOGIN	COM2CC		DUMMY ENTRY - ONLY NAME USED	COMTBL	49
LGOTORD	EQU	*	FWA OF LOGOUT ENTRY	IN41353	6
LOGOUT	COM2CC		DUMMY ENTRY-ONLY NAME USED	IN41353	7
				COMTBL	50
PRIOR	REMOTE	P=NO, MP=3, REMORD=2, SUBORD=0, QP=PARM3, MQ=OUTPUT		COMTBL	51
REDUCE	COM2CC	L=YES, P=NO, B=NO, MP=1		COMTBL	53
REVERT	COM2CC	L=NO, P=NO, B=YES, MP=1		COMTBL	54
SAVEFL	COM2CC	L=YES, P=NO, B=NO, MP=1		COMTBL	56
SWITCH	COM2CC	L=YES, P=YES, B=NO, MP=1		COMTBL	58
SIGNON	COM2CC	L=NO, P=NO, B=NO, SLST=YES		IN5A121	7
				COMTBL	59
VERB7	EQU	*		COMTBL	60
				COMTBL	61
COMMENT	COM2CC	L=NO, P=YES, B=NO		IN41998B	2
MESSAGE	COM2CC	L=NO, P=YES, B=NO		IN40844C	14
				COMTBL	66
TBLEND	EQU	*	LWA+1 OF TABLE = LWA OF VERB7 SECTION	COMTBL	67

Figure II-28-1. 2CS Release Values and Update Identifiers

*	MUJTABL	FOLLOWS.	ALL DEFINED MUJ'S ARE INCLUDED.	THEY MUST	1QP	594
*		PHYSICALLY OCCUR	IN THE SAME ORDER AS THEY DO	IN 1CI'S TABLE.	1QP	595
*					1QP	596
	MUJTABL	BSS	0	DEFINE BEGINNING OF TABLE	1QP	597
		MUJTBL		EDITOR, 40000, 0, 2, 1	1QP	598

Figure II-28-2. 1QP Release Values and Update Identifiers

## EDITOR INSTALLATION PARAMETERS

EDITOR uses two common decks, IPFTN (FORTRAN) and IPCOM (COMPASS), to contain installation parameters. Generally, a change to one common deck requires a corresponding change to the other. With the exception of arrays which must be dimensioned for FORTRAN in common deck IPFTN, the values of installation parameters are not defined in IPFTN. IPFTN merely allocates storage for these definitions. The definitions are DATA statements in the BLOCK DATA subprogram IPFILL.

IPCOM contains EQU's which define the installation parameters. Since many parameters are of such a nature that a change in one implies a change of another, a dependency chart is included (table II-28-2) to aid the installation.

Following is a summary of the steps to be taken to change an EDITOR installation parameter.

1. Change the DATA statement in IPFILL or the EQU in IPCOM, or both, as indicated by the parameter description.
2. Consult the dependency chart (table II-28-2) for any dependent installation parameters that require change, and change them as in step 1.
3. Consult the dependency chart (table II-28-2) for dimensions of arrays in IPFTN. If they are affected, change them as indicated in table II-28-3.

Additionally, EDITOR has the installation parameter IP.FTNTS defined in common deck INTCOM (refer to INTERCOM Common Deck Settings).

Any changes which cause the size of the EDITOR to increase may require an increase in the field length defined for EDITOR in the MUJTABL for IQP. The following list shows the release values and UPDATE identifiers for IPFILL, IPCOM, and IPFTN.

### IPCOM

* THE FOLLOWING SYMBOLS MUST BE DEFINED FIRST, SINCE THEY ARE USED TO			IPCOM	3	
* DEFINE OTHER SYMBOLS BELOW			IPCOM	4	
NPRUS	EQU	2	NUMBER OF PRUS IN EDIT FILE *WINDOW*	1410095	3
NTBSMAX	EQU	10	MAX. NUM. OF TAB SETTINGS ALLOWED	1410095	4
NSINDEX	EQU	20	SIZE OF EDITFIL INDEX	1410079	1
* INSTALLATION PARAMETERS			IPCOM	86	
* SEE ALSO- ADDITIONAL PARAMETERS DEFINED AT VERY BEGINNING OF IPCOM			IPCOM	87	
NBBS	EQU	2	NUMBER OF BIG BUFFERS	IPCOM	88
NSBB	EQU	257	SIZE OF BIG BUFFER (WITHOUT FET)	1410095	5
NPBS	EQU	3	NUMBER OF POOL BUFFERS	IPCOM	90
NUSERS	EQU	30	MAXIMUM NUMBER OF ATTACHED USERS	IPCOM	91
NEDFETS	EQU	10	MAXIMUM NO. OF EDIT FILE FETS	IN40815	42
NUAS	EQU	3	NUMBER OF USER AREAS	IPCOM	93
NSUA	EQU	JOPRU+NPRUS*64		IN40815	43
* EXCLUDES THE TAB POSITIONS, THE INDEX, AND THE RJLNKS AREA)			IPCOM	95	
NSRJLNK	EQU	10	SIZE OF RU LINKAGE AREA	IPCOM	97
NUASIZE	EQU	JRULNKS+NSRJLNK		IN40815	44
* DEBUG OPTION - IF DEBUG EQU 1, DEBUGGING CODE IS ASSEMBLED			IPCOM	98	
NDEBUG	EQU	0	NO DEBUGGING CODE	IPCOM	99
			IPCOM	100	



IPFILL

C	IMPORTANT NOTE - THE VALUE OF THE FOLLOWING VARIABLES IS DEPENDENT	IT30062	31
C	ON THE VALUE OF THE INSTALLATION PARAMETER NPRUS -	IT30062	32
C	NSUA,	IT30062	33
C	JTABS, UNDXHDR, JINDEX, JRJLNKS	IT30062	34
C	IF NPRUS IS NOT EQUAL TO ONE, THE FOLLOWING MODIFICATIONS MUST BE	IT30062	35
C	MADE. FOR EACH OF THE VARIABLES LISTED ABOVE, ADD	IT30062	36
C	THE VALUE (NPRUS-1)*64.	IT30062	37
	DATA JTEXT1/0/,JTEXT2/2/,JFLAGS1/4/,JFLAGS2/5/,JPBW/6/,JLNBUF/17/,	IN40815	741
	X JCPRU/69/,UTABS/197/,JNDXHDR/199/,JINDEX/200/,JRJLNKS/220/	IN40815	742
	DATA NLINE/6L000100/,NINCR/10/,NUAS/3/	EDITOR	4251
	DATA NBBS/2/,NPBS/3/,NUSERS/30/,NEDFETS/10/,	IN40815	746
	X NSUA/197/,NSINDEX/20/,NSRJLNK/10/,NTBSMAX/10/,XNPCENT/0.9/,	IN40815	747
	X NUASIZE/230/,NPRUBUF/128/	IN40815	748
C	SEE NOTE ABOVE IF NPRUS IS CHANGED.	IT30062	38
	DATA NPRUS /2/	I410095	8
	DATA NTABFTN/1L;/,NTBSFTN/5/,NCHFTN/72/,	F7840D4.17	
	X FTNTABS/00070012001500200023B/	F7840D4.18	
	DATA NTABCOM/1L;/,NTBSCOM/3/,NCHCCM/72/,	EDITOR	4256
	X COMTABS/00130022004400000000B/	EDITOR	4257
	DATA NTABCOB/1L;/,NTBSCOB/5/,NCHCCB/72/,	EDITOR	4258
	X CCBTABS/00100014002000240030B/	EDITOR	4259
	DATA NTABALG/1LS/,NTBSALG/5/,NCHALG/72/,	IT30125	1
	X ALGTABS/00070012001500200023B/	EDITOR	4261
	DATA NCHBAS/150/	IN41370	113
	DATA NTABDEF/1L;/,NTBSDEF/5/,NCHDEF/72/,	F7840D4.19	
	X DEFTABS/00070012001500200023B/	F7840D4.20	

IPFTN

C	THIS COMMON DECK CONTAINS ALL INFORMATION CONCERNING THE FORMAT OF	IPFTN	3
C	THE EDITOR COMMON AREA (ECA) AND INSTALLATION PARAMETERS	IPFTN	4
C	THIS STATEMENT DEFINES THE EDITOR COMMON AREA	IPFTN	8
C	ALL OF THE CELLS IN COMMON BLOCK /ECA/ ARE FILLED BY (ECAFILL)	IPFTN	9
	COMMON /ECA/ MLRROR, MEM(1), MRA,	IN40815	5
	X MPTR1, MPTR2, MCNT,	IPFTN	11
	1 MIC1, MCASE, MSTATE, MCASE2, MSTATE2, MUNMBR, MOASUB, MEFPTR, MACTN,		
	2 MCMLPT, MTEBCNT, MTEB(S2), MBBMA(2), MPBMA(3), MBBS(526), MPBS(384),	I410095	1
	3 MEDFST(80), MMUJTBL(159), MPBWD, MBBWD, MUAS(690)	IN40815	7
	X, IEDFET, ITPRUS, IWDSWRT, IUFRUS	IN40815	8
C	NUMBERS OF WORDS IN USER AREA (RELATIVE TO ZERO)	IPFTN	37
	COMMON/ IPFTN/ JTEXT1, JTEXT2, JFLAGS1, JFLAGS2, JPBW, JLNBUF, JCPRU,	IPFTN	38
	X JTABS, JNDXHDR, JINDEX, JRJLNKS	IPFTN	39
C	NUM. OF WDS IN EDIT FILE WINDOW- INITIALIZED AT BEGINNING OF EDITOR	IPFTN	62
	COMMON /IPFTN/ NPRUBUF	IPFTN	63
C	INSTALLATION PARAMETERS	IPFTN	64
	COMMON/ IPFTN/ NLINE, NINOR, NBBS, NP8S, NUAS, NUSERS,	IN40066	1
	X NSUA, NSINDEX, NSRJLNK, NTBSMAX, XNPCENT, NPRUS, NUASIZE, NEDFETS	IN40815	12
C	(FORTRAN) FORMAT	IPFTN	67
	COMMON/ IPFTN/ NTABFTN, NTBSFTN, NCHFTN, FTNTABS(1)	IPFTN	68
C	(COMPASS) FORMAT	IPFTN	69
	COMMON/ IPFTN/ NTABCCM, NTBSCOM, NCHCOM, COMTABS(1)	IPFTN	70
C	(COBOL) FORMAT	IPFTN	71
	COMMON/ IPFTN/ NTABCOB, NTBSCOB, NCHCOB, COBTABS(1)	IPFTN	72
C	(ALGOL) FORMAT	IPFTN	73
	COMMON/ IPFTN/ NTABALG, NTBSALG, NCHALG, ALGTABS(1)	IPFTN	74
C	(BASIC) FORMAT	IPFTN	75
	COMMON/ IPFTN/ NCHBAS	IPFTN	76
C	DEFAULT FORMAT	IPFTN	77
	COMMON/ IPFTN/ NTABDEF, NTBSDEF, NCHDEF, DEFTABS(1)	IPFTN	78

In table II-28-1, -\* in the Range column indicates where a parameter has essentially no absolute upper limit. The installation determines the practical upper limit based on considerations such as EDITOR size and expected number of users.

TABLE II-28-1. EDITOR INSTALLATION PARAMETERS

Parameter Name	Defined In		Description	Range	Release Value
	IPFILL	IPCOM			
NLINE	X		Default first line number for CREATE, EDIT, RESEQ	6L000001 to 6L999999	6L000100
NINCR	X		Default line number increment for ADD, CREATE, EDIT, RESEQ	1-999998	10
NUAS	X	X	Number of user area buffers	1-* Large number decreases response time if there are many users	3
NBBS	X	X	Number of big buffers (used for EDIT, SAVE, RUN)	1-* Increase if many EDITS, SAVES, RUNs anticipated	2
NPBS	X	X	Number of pool buffers. Each is 64*NPRUS words	2-* Increase when heavy file modifications or long text lines expected, generally NPBS>NUAS	3
NUSERS	X	X	Maximum number of users simultaneously using EDITOR	1-* Vary with expected usage of EDITOR	30
NPRUS	X	X	Number of 64-word PRUs in one block in edit file  If NPRUS is larger than 12D, EDITOR will not use the CMU hardware, regardless of the setting of NOS/BE 1 installation parameter IP.CMU.	1-* Large number decreases response time for commands which process large files, but it also increases amount of central memory required for EDITOR by 64 words for each pool buffer and 64 words for each user area buffer	2
NSUA	X	X	Size of user area; must be modified in IPFILL if NPRUS is changed. NSUA=69+ 64*NPRUS. Size does not include portion of user area used for tabs, return jump links, and edit file index	133-*	197
NUASIZE	X	X	Size of user area including areas for tabs, return jump links and edit file index.	133-*	230
NPRUBUF	X		Number of words in one edit file block. Must be 64*NPRUS	64-*	128
JTABS	X		Number of word in user area which holds tab values; must be modified in IPFILL if NPRUS is changed. JTABS= 69+64*NPRUS	131-*	197

TABLE II-28-1. EDITOR INSTALLATION PARAMETERS (Contd)

Parameter Name	Defined In		Description	Range	Release Value
	IPFILL	IPCOM			
JNDXHDR	X		Number of index header word in user area; must be modified in IPFILL if NPRUS is changed. JNDXHDR= JTABS+(NTBSMAX+4)/5	132-*	199
JINDEX	X		Number of first word in edit file index in user area; must be modified in IPFILL if NPRUS is changed. JINDEX= JNDXHDR+1	133-*	200
JRJLNKS	X		Number of first word in return jump link area in user area; must be modified if NPRUS is changed. JRJLNKS= JINDEX+NSINDEX	153-*	220
NSINDEX	X	X	Number of index entries for each user's edit file	1-* Increase for editing very large files	20
NTBSMAX	X	X	Maximum number of tab settings permitted by FORMAT command	1-509 Must be > NTBSFTN, NTBSCOM, NTBSCOB, NTBSALG, NTBSDEF	10
XNPCENT	X		Percent to which each block of user's edit file is filled by EDIT (Padding factor)	.01-1.00 Decrease if heavy file modification is expected	.90
NTABFTN	X		FORTRAN tab character	1 LA-1 L;	1 L;
NTABCOM	X		COMPASS tab character	1 LA-1 L;	1 L;
NTABCOB	X		COBOL tab character	1 LA-1 L;	1 L;
NTABALG	X		ALGOL tab character	1 LA-1 L;	1 L\$
NTABDEF	X		Default tab character	1 LA-1 L;	1 L;
NTBSFTN	X		Number of FORTRAN tabs defined	0-509	5
NEDFETS	X	X	Number of FETs used to attach a user's editfile.	1-*	10
NTBSCOM	X		Number of COMPASS tabs defined	0-509	3
NTBSCOB	X		Number of COBOL tabs defined	0-509	5
NTBSALG	X		Number of ALGOL tabs defined	0-509	5
NTBSDEF	X		Number of default tabs defined	0-509	5
NCHFTN	X		Maximum number of characters in FORTRAN line	1-510	72
NCHCOM	X		Maximum number of characters in COMPASS line	1-510	72

TABLE II-28-1. EDITOR INSTALLATION PARAMETERS (Contd)

Parameter Name	Defined In		Description	Range	Release Value
	IPFILL	IPCOM			
NCHCOB	x		Maximum number of characters in COBOL line	1-510	72
NCHALG	X		Maximum number of characters in ALGOL line	1-510	72
NCHDEF	X		Maximum number of characters in default format	1-510	72
NCHBAS	X		Maximum number of characters in BASIC line		150
FTNTABS	X		Consecutive stream of bits, each 12 define a tab position for FORTRAN format. Must be ascending order	1-511 (each tab)	00070012001500200023B
COMTABS	X		Same as above, for COMPASS	1-511	00130022004400000000B
COBTABS	X		Same as above, for COBOL	1-511	00100014002000240030B
ALGTABS	X		Same as above, for ALGOL	1-511	00070012001500200023B
DEFTABS	X		Same as above, for default format	1-511	00070012001500200023B
NSBB		X	Size of big buffers used for EDIT, SAVE, RUN (does not include FET)	64-* Increase for very large files	157
NDEBUG		X	Flag controls debugging mode.  (Refer also to multiuser job installation parameter MDEBUG)	0 = off 1 = on	0

TABLE II-28-2. EDITOR DEPENDENCIES

Changed Parameter	Check Parameters in IPFILL and/or IPCOM	Check Arrays in IPFTN
NLINE NINCR NUAS NBBS NPBS NUSERS† NSINDEX NTBSMAX XNPCENT NTABFTN NTABCOM NTABCOB NTABALG NTABDEF NTBSFTN NTBSCOM NTBSCOB NTBSALG NTBSDEF NCHFTN NCHCOM NCHCOB NCHALG NCHDEF NCHBAS FTNTABS COMTABS COBTABS ALGTABS DEFTABS NSBB NDEBUG NPRUS	JRJLNKS JNDXHDR, JINDEX, JRJLNKS  NTBSMAX, FTNTABS NTBSMAX, COMTABS NTBSMAX, COBTABS NTBSMAX, ALGTABS NTBSMAX, DEFTABS  NTBSFTN NTBSCOM NTBSCOB NTBSALG NTBSDEF  JTABS, JNDXHDR, JINDEX, JRJLNKS, NSUA	MMUJTBL, MUAS MBBS, MBBMA MPBS, MPBMA, MMUJTBL MMUJTBL MUAS MUAS  FTNTABS COMTABS COBTABS ALGTABS DEFTABS  FTNABS COMTABS COBTABS ALGTABS DEFTABS MBBS MDEBUG † † MPBS, MUAS
† When NUSERS is increased, the user should also consider changing the size of the TERMIN and TERMOUT tables in the muj subroutine MUJSUBS. Refer to the INTERCOM 4 Multiuser Job Capability Programming System Bulletin under the heading Changing Size of TERMIN and TERMOUT. † † Refer to Multiuser Job Installation Parameters..		

TABLE II-28-3. EDITOR ARRAY DIMENSIONS IN IPFTN

Array Name	Usage	Array Dimension
FTNTABS	FORTRAN tabs	(NTBSFTN+4)/5
COMTABS	COMPASS tabs	(NTBSCOM+4)/5
COBTABS	COBOL tabs	(NTBSCOB+4)/5
ALGTABS	ALGOL tabs	(NTBSALG+4)/5
DEFTABS	Default tabs	(NTBSDEF+4)/5
MMUJTBL	Storage needed by muj subroutine tables	4*NUMERS + NBBS + NEDFETS + NUAS + 6
MUAS	User area buffers  May never exceed 4095 decimal	NUAS* (size of full user area) where:  (size of full user area) (NTBSMAX+4)/5 + NSINDEX + 1 + NSUA + NSRJLNK  Note: NSRJLNK should not have to be changed by an installation
MBBS	Big buffers	NBBS*NSBB + NBBS*6
MPBS	Pool buffers	NPBS*64*NPRUS
MBBMA	Big buffer management area	NBBS
MPBMA	Pool buffer management area	NPBS

EDITOR DEBUG CODE

If EDITOR encounters hardware and/or software problems, a diagnostic printout is produced. If the problem is considered fatal, all EDITOR users are detached. The content of the diagnostic printout depends on the error encountered and the setting of NDEBUG. In any event, the diagnostic printout should accompany any PSR relating to a MUJ SYSTEM ERROR. Refer also to MDEBUG in the following subsection.

MULTIUSER JOB INSTALLATION PARAMETERS

The multiuser job (muj) subroutines use two common decks, MUJCOM and CMUJCOM. Both contain storage allocation for an array, ECSBUF. The MUJCOM deck in FORTRAN code contains a DIMENSION statement; the CMUJCOM deck in COMPASS code contains a BSS statement. This array is used by the muj peripheral processor routines, FAD, to read information from extended core storage (ECS). Array length must be (n\*64+1) central memory words. The value of n can be selected by the installations, depending on the expected use of ECS for storage of user swap files (if ECS is used, n should be at least 2) and on the number of local files allowed for an INTERCOM user. As a guide, n may be increased by one for each 20 local files allowed per user. The upper limit for n is dependent on the amount of storage used for the ECS buffer in the muj, and the size of the swap buffer in FAD.

The peripheral processor routine FAD contains two parameters relevant to allocation of space for ECSBUF. ECSBFLN (near FAD.659) is a COMPASS EQU instruction. It must be equated to the number of central memory words in the ECSBUF array. SWAPBF (near FAD.650) is a table FAD uses to read the ECSBUF array into PP memory. The value of ECSBFLN, and thus the size of the ECSBUF array in MUJCOM and CMUJCOM, must not be greater than 1 + (length of SWAPBF)/5.

Symbol MDEBUG in common deck CMUJCOM controls muj debugging code (0=off, 1=on). It should be set to 1 if the EDITOR installation parameter NDEBUG is set to 1.

In the routine MUJFILL, the two constants NACOUNT and THRSHLD control the accounting of muj time. The value of NACOUNT determines how frequently the accounting information for a muj is obtained from the system and distributed to users attached to the muj. NACOUNT is the maximum number of user switches performed on any given user before accounting is done. Accounting is always performed on user exit from the muj. NACOUNT must be set greater than or equal to 1 and defaults to 50 decimal. The value of THRSHLD determines the minimum number of CP seconds accumulated before accounting is posted to the user. As THRSHLD is set to smaller values, accounting is more accurate, overhead is increased, and THRSHLD has more meaning. THRSHLD defaults to 5 decimal.

## INSTALLATION PROCEDURES

Installation job decks PL14I and PL14E can be obtained from the Installation Decks program library, using the procedure outlined in part I, section 1 of this document.

Deck PL14I assembles the released program library adding the created binary to the PL tape as supplemental files. The release tape does not contain assembled binary. Deck PL14E uses EDITLIB to enter the binary created by deck INTCM1 into the running system.

Deck PL14O, applicable only in the user library method of installation, allows regeneration and replacement of absolute overlays in the user library plus creation of a new PL14 tape.

Deck PL14E suggests CM residency for selected PP routines. Sites having ECS may wish to move some of these PP routines to ECS by employing the method discussed in the System ECS Resident Routine and Library portion of part II, section 1. Once PL14E has been run, job DST3 can be run to capture a deadstart tape containing INTERCOM. Decks PL14E and DST3 need not be run if the user library installation process is being followed.

No INTERCOM PP programs are required to be CM resident. However, to increase product performance, installation jobs PL14E and DST1 contain EDITLIB MOVE directives to force into CM residence some driver overlays, 3TT and its overlays, 1CI and two of its overlays, and 1QP. This group of routines and overlays requires 4500 octal words. Additionally, 2ND, 3ND, 4ND, 5ND, 6ND, 7ND, and 8ND should be CM resident to prevent INTERCOM restarts in situations (usually caused by error conditions) where the disk or ESC access to read the INTERCOM driver is not available for periods of 3 seconds.

INTERCOM in an idle state uses 4500 octal words for multiplexer tables, PRUB buffers, and minimum empty buffer chains.

## NPU PROGRAMS

The INTERCOM 2550 Front End NPU initializer uses the NPU multiplexer subtables to determine which variants of the NPU programs to load before the NPU driver is brought up. INTERCOM assumes the proper variants are available on the system library and are disk-resident. The general format of NPU load module names is 0Dy, where y is a value computed from the following values assigned to the line protocols

1	Mode 3
2	Mode 4
4	IBM 2780/3780
8	HASP

Add the values for each different type of line configured. Convert the total (1 to 15) to hexadecimal (1 to F).

### Example

A 2550 configuration includes 100 mode 3 lines only. The load module name consists of 0D1, since the value of mode 3 lines is 1.

Two additional types of load modules are used.

0D0	Micro memory module
0DZ	Bootstrap dump routine

All of these programs are available as part of a separate release library for CCI 3 that includes the NPU programs. NPU binaries can be added to the running system using the following job.

```
Job statement.  
EDITLIB(SYSTEM)  
7/8/9  
READY(SYSTEM,OLD)  
REPLACE(*,INPUT)  
COMPLETE.  
ENDRUN.  
7/8/9  
Binary decks of NPU programs  
6/7/8/9
```

Section 29 of this part describes CCI 3 installation in detail.

#### MUJ SYSTEM ERRORS

INTERCOM multiuser jobs (for example, EDITOR), upon encountering hardware and/or software errors, produce diagnostic dumps. These dumps contain a header MUJ SYSTEM ERROR xx. This message is sent to the system dayfile and to each user currently using the muj. Error codes and their significance are described in the NOS/BE Diagnostic Handbook.



## PASSWORD FILE CREATION

Access to the INTERCOM system is controlled by passwords. The user must specify a valid password to log in to the INTERCOM system. Two types of passwords exist: restricted and unrestricted passwords.

With restricted passwords, when logging in, the user must specify a valid username associated with the given password. The installation defines valid username/password combinations. A user id (two alphanumeric characters) is assigned by the PASSWRD utility, and it is permanently associated with the username/password. This user id is assigned from a pool of available user ids; it is marked as available again only when the username/password is deleted.

With unrestricted passwords, the user may specify any username when logging in. The username is not validated. However, when a user first logs in under a given username, a user id is associated by the LOGIN utility with that username/password combination. Thereafter, this user id is associated with the username/password combination, until the username/password is deleted from the system.

Through the INTERCOM routine PASSWRD, the installation defines valid restricted username/password combinations and valid unrestricted passwords and accounting values to be associated with the username/passwords or passwords. PASSWRD must be called from a data deck submitted to the central site as a batch job. The routine creates two permanent files (or edit existing files). One file, with the permanent file name INTERCOMPASSWORDS, contains all unrestricted passwords, all restricted username/passwords, and all accounting information. The other file, with the permanent file name INTERCOMUNRESTRICTED, contains a bit map defining assigned user ids; it also contains all unrestricted username/password combinations. Installations with many users should do the following.

- Instruct users of unrestricted passwords always to use the same character string for username when logging in.
- Make use, on a regular basis, of the editing facilities in PASSWRD to delete all unrestricted usernames, and so on, freeing user ids.

While a user is in the process of logging in, he is assigned a temporary id. Temporary ids begin with a special character.

The following deck structure can be used to run the PASSWRD routine, creating a password permanent file.

```
Job statement.  
PASSWRD.  
7/8/9  
NEW  
ADD  
.  
.  
.  
6/7/8/9
```

The following deck structure can be used to modify existing password permanent files.

```
Job statement.  
PASSWRD.  
7/8/9  
OLD  
ADD or  
DEL  
.  
.  
.  
6/7/8/9
```

This mode of PASSWRD operation updates the existing permanent files by adding new or deleting old entries. If both files do not exist, a PF ERROR=12B aborts the run.

To protect against unauthorized modification of the password files, the PASSWRD utility requests permission from the console operator before any modifications are made.

Between the NEW (or OLD) statement and the 6/7/8/9 statement appear the parameter statements which specify the new entries or the editing requirements. After a NEW statement, only ADD parameter statements may appear; after an OLD statement, either ADD or DEL parameter statements may appear. The ADD statement creates a new entry, or replaces an old entry which has the same username/password. The DEL statement deletes one or more entries. The NEW statement may be used to delete existing files entirely and to construct new ones.

The format for an ADD parameter statement is the following.

ADD U=username,P=password,F=length,T=time,A=acclvl,N=nfiles,E=ecsfl

- username      Username (1 to 10 alphanumeric characters) must be specified for restricted passwords; it must be blank or omitted for unrestricted passwords.
- password      Password (1 to 10 alphanumeric characters) must be specified. It must be the only unrestricted password of this name defined by the installation. If it is restricted, it must be the only username/password of this particular combination defined by the installation. (If the password or username/password have been previously defined, the ADD card functions as a replace.)
- length        Maximum field length available to the user (1 to 6 octal digits). If blank or omitted, 60000 octal CM words are assumed. This value must not exceed IP.MFL.
- time          Time limit for user's session (1 to 4 octal digits, also defines the maximum ETL for individual jobs). If blank or omitted, 500 octal seconds are assumed.
- acclvl        Access level/permission bits for the user (0-3777<sub>8</sub> range). This value defines which programs the user can access. If blank or omitted, an access level of 5 is assumed (dependent on IP.IACES setting in common deck INTCOM).
- nfiles        Number of files this user is permitted to attach as local files at any one time (1 to 2 octal digits). If blank or omitted, 24 (octal) files are allowed. This value must not exceed 76<sub>8</sub>.
- ecsfl        Maximum ECS field length available to the user in multiples of 1000B (1 to 4 octal digits). If blank or omitted, zero is assumed. This value must not exceed IP.MECS.

All parameters start after column 4 on the ADD and DEL statements. They can be specified in any order and should be separated by delimiters (special characters).

The DEL statement is used to delete one or more entries from one or both of the permanent files. It has two formats.

DEL U=username,P=password

DEL I=id

- username      May take three forms: 1 to 10 alphanumeric characters, blank, or the character string \*NAMES. If the first form is used, the username/password combination (restricted or unrestricted) is deleted; and the user id becomes available. If the second form is used, all entries in the two files with the given password are deleted. All user ids associated with these entries will become available; the password will no longer be defined. The third form may be used only if the specified password is unrestricted. All entries in the unrestricted password file with the given password will be deleted, and the associated ids will be made available. The password will still be defined.
- password      Password to be processed. Whether an unrestricted password is deleted or not depends on the username parameter. If password is \*NAMES, all usernames for all unrestricted passwords are deleted from the permanent files and the user ids for these usernames become available. The unrestricted passwords will still be defined.
- id             User id; may be used as a shorthand notation to specify the username/password associated with this user id. The given username/password entry (restricted or otherwise) is deleted and the user id becomes available. If the password is unrestricted, it will still be defined.

SCED INSTALLATION PARAMETERS

When a multiuser job which uses SCED is installed, default parameter values in SCED should be changed to reflect the requirements of the COBOL program involved. A value should be changed by deleting the default definition macro call and replacing it with a call to the SCED macro with the new parameter value. All macros are required.

Example

```
*D      SCED.233      Deletes MAXUSR parameter
         MAXUSR 10      Replaces MAXUSR with new value
```

The SCED macro (parameter) calls are described in detail in the INTERCOM 5 Multiuser Job Capability Reference Manual.

<u>Parameter</u>	<u>Default Value</u>	<u>Line to Replace</u>
MAXUSR	30	SCED.233
USAREA	2,214	SCED.234
NUMINT	40	SCED.235
DEFBUF	_____	No parameters
		No need to replace
OUTBUF	4,45	SCED.237, SCED.238
	4,144	

VERIFICATION PROCEDURE

INTERCOM is brought to control point zero when INTERCOM is entered at the console after the operator has entered the time.

The verification procedure cannot proceed unless a permanent file has been established containing the user passwords.

The following sample from an interactive terminal session indicates if INTERCOM is installed correctly. The underlined characters are typed by the user.

CONTROL DATA INTERCOM 5.0

DATE 06/27/77  
TIME 09.27.22.

PLEASE LOGIN

LOGIN

ENTER USER NAME- THOBBIE

ENTER PASSWORD- MYPASSWRD

06/27/77 LOGGED IN AT 09.28.46.  
WITH USER-ID D3  
EQUIP/PORT 47/04

COMMAND- SITUATE

USERS WITH SAME PASSWORD

D3-THOBBIE

OTHERS

B6-HALLA	ER-IPRICE	FL-ALL155
FM-ALL156	BC-OPS	FI-TAYLOR
F3-ZEE	BA-4800BAUD	BB-4801BAUD
BD-HSBT	BE-MSBT	GU-SVLANX
GY-CHESLEY	HN-EBROTH	G4-JGM

BATCH TERMINALS

AS-MODE 4A AU-MODE 4C AV-MODE 4A

AF-HASP AG-MODE 4A

COMMAND- ASSETS

ASSETS OF D3 AT 09.30.11.  
EQUIP/PORT 47/04  
FILE QUOTA 20  
FILES IN USE 0  
MAX FL 0077700  
TIME LIMIT 7000  
CP TIME .164  
COMMAND- ETL,100

COMMAND- MAP,ON

COMMAND- ASSETS

ASSETS OF D3 AT 09.31.00.  
EQUIP/PORT 47/04  
FILE QUOTA 20  
FILES IN USE 0  
MAX FL 077700  
TIME LIMIT 7000  
ETL 0100  
MAP ON  
CP TIME .174  
COMMAND- FILES

NONE

COMMAND- LOGOUT

CPA .198 SEC. .198 ADJ.  
SYS TIME 1.159  
CONNECT TIME 0 HRS. 5 MIN.  
06/27/77 LOGGED OUT AT 09.31.49.



RELEASE DESCRIPTION

Communications Control INTERCOM (CCI) Version 3 is the software and loadable controlware that supports the 255x Network Processing Unit (NPU) as a front end to INTERCOM Version 5 on CYBER 170, CYBER 70, and 6000 Series computers. The CCI binary load modules reside in the NOS/BE 1 PPU library so they can be loaded into the 255x by INTERCOM.

Two release tapes are associated with CCI 3. PL99A consists of the CCI source program file (MUX firmware source and Post Link Editor initialization directives), the System Creation File (SCF), the binary load and listing files for the MUX firmware, and boot dump programs. PL99B consists of CCI 2550 binary macro load file, two intermediate files, the object file (LGO), and four build listings files.

CCI installation creates three downline load modules that resemble PPU binaries. The format of the downline load module names is 0Dy, where y is as follows.

- 0 For the micro memory load module
- Z For the boot dump load module
- 1-F For the NPU load module, representing the sum of the assigned values for TIPs defined as follows.
  - ASYNC (MODE 3) 1
  - MODE 4 2
  - TIP780 4
  - HASPTIP 8

Table II-29-1 summarizes the module types and conditions under which each type can or must be created. Refer to the CCI Reference Manual for descriptions of the functions performed by each module type.

TABLE II-29-1. CCI MODULE REQUIREMENTS

Module Type	Number Required	Loaded Into	Installation Deck(s)	Comments
Dump/bootstrap 2550	One required if 2550 is in use	All 2550's in network	PL99AI1	Required name is 0DZ.
Phase 1 load 2550	One required if 2550 is in use	All 2550's in network	PL99AI1	Micromemory (MUX firmware); required name is 0D0.
Phase 2 load 2550	One required if 2550 is in use	2550 for which module is configured	PL99AI2 PL99AVI	Macromemory; required name is 0Dx, where x can be 1 through F.

## INSTALLATION OVERVIEW

The PL99AI1 installation job updates the two program libraries (PLs) on PL99A using the PSR batched corrective code and user/critical code on file OSMINIT. It also produces the phase 1 load (micromemory) and dump/bootstrap modules for the 2550. The PLs on PL99A are also used by installation jobs PL99AI2 and PL99AVI to produce all other modules.

The installation jobs use the following procedure to integrate PL99AI1, PL99AI2, and PL99AVI.

1. The job updates the SCF PL to produce a compile file. This file contains Update directive records as well as directives used by MPLIB, MPLINK, and MPEDIT later in the job when creating the CCI module(s).
2. The job updates the base PL with the Update directives on the SCF compile file to extract the decks needed to produce a particular CCI module depending upon the job being run.
  - a. PL99AI2 produces decks containing input to PASCAL, MASSEM, ASSEM, MPEDIT, and MPLIB.
  - b. PL99AVI only produces decks containing input to MPLIB, MPLINK, and MPEDIT.

After integrating the PLs and extracting the appropriate source decks for input to the cross processors, each installation job creates one or more CCI modules. PL99AI2 also produces an object file that is cataloged and copied to tape PL99B. From these object files, PL99AVI creates 2550 variants configured for individual NPUs. To achieve maximum flexibility when creating variants in this manner, the object file created by PL99AI2 includes all options that could be included in any variant used in the network. This mechanism is provided because the assemblies and compilations performed by PL99AI2 require significant amounts of time, whereas the object file manipulations performed by PL99AVI are relatively fast.

## RELEASE MATERIALS

Release materials consist of three 7- or 9-track system standard labeled magnetic tapes. The PL99A tape contains the following files.

<u>File Number</u>	<u>Record Number</u>	<u>File content</u>	<u>Name</u>	<u>Comments</u>
1	1	CCI program library (including MUX firmware source and MPEDIT initialization directives)	CCI30BLD	Update PL
2	1	Build input PL	CCI30BUILD	Update PL
3	1	2550 Micro Load (MUX Firmware 1412)	MPPPU0D0	PPU format downline load file 0D0
4	1	2550 Boot Dump	MPPPU0DZ	PPU format downline load file 0DZ
5	1	2550 Firmware ZAPMP	ZAPMP0D(1-F)	Core image provided by MPEDIT
6	1	2550 Micro Load List	LIST	Print file
7	1	2550 Boot Dump List	LIST	Print file

The PL99B tape contains the following files.

<u>File Number</u>	<u>Record Number</u>	<u>File Content</u>	<u>Name</u>	<u>Comments</u>
1	1	2550 Macro Load	MPPPU0Dx	PPU format downline load file 0Dx, where x can be 1 through F
2	1	ZAPMP	ZAPMP0Dx	Core image provided by edit phase; x can be 1 through F
3	1	Symbol table	SYMTAB0Dx	Symbol table provided by link phase; x can be 1 through F
4	1	Macro object file	LGO	Object file
5	1	Assembly list	LIST	Print file
6	1	PASCAL source listing	LIST	Print file
6	2	PASCAL object listing	LIST	Print file
7	1	MPLINK list	LIST	Print file
8	1	MPEDIT list	LIST	Print file

## HARDWARE CONFIGURATION

The minimum hardware configuration to build CCI 3 requires a job field length of 77,000 octal words while running NOS/BE 1. (Running PASCAL with a field length of 77K can require a long time to run on a busy system; using the 125K version is much faster.)

### CCI HARDWARE REQUIREMENTS

The minimum equipment configuration required to execute CCI consists of the following.

- 1        2550-2 or 2551-1 Network Processing Unit which includes the following.
  - 1        Multiplexer Loop Interface Adapter
  - 1        Loop Multiplexer
  - 1        Cyclic Encoder Board
  - 1        CDC CYBER Communications Coupler
- 1        16K Memory Unit with 2550-2 Processor
- 1        Communications Line Adapter from either of the following.
  - 2560-1 Synchronous CLA
  - 2561-1 Asynchronous CLA
- 1        32K Additional Memory Unit
- 1        16K Additional Memory Unit

### NOTES AND CAUTIONS

The communications line adapter slots in the loop multiplexer should be assigned in order of decreasing line transmission speeds. For example,

9600 bps line	Slot 1 (left-most slot)
9600 bpi line	Slot 2
2400 bps line	Slot 3
300 bps line	Slot 4
150 bps line	Slot 5

### INSTALLATION PARAMETERS

Parameters that can be adjusted during the creation of CCI software load files are of three types.

- Type 1    MPEDIT constants†
- Type 2    Update DEFINE directives used during compile file creation
- Type 3    SCF build input parameters†

† Any changes to type 1 or type 3 parameters require modifications to the installation deck oldpl.



The following installation parameters can be modified in common decks PL99DEFS and PL99IN with their acceptable and default values. Numeric values preceded with a \$ are given in hexadecimal.

All type 1 statements end with a semicolon; they can be followed by a comment which is preceded by a right arrow and followed by a down arrow.

#### TYPE 1 STATEMENTS

The syntax for items of type 1 is as follows.

name = value; → COMMENTS ↓

There are no column restrictions.

Example

/C4LCBS=80; →NUMBER OF LINES ↓

The following are MPEDIT constants.

<u>Card Identifier</u>	<u>Name</u>	<u>Description</u>	<u>Acceptable Values</u>	<u>Default Value</u>	<u>Restrictions</u>
ZEXUSR.19	VM4FAIL	Number of 1/2 second intervals between polls for a failed mode 4A terminal.	1-63	5	
ZEXUSR.20	VM4CFAIL	Number of 1/2 second intervals for a failed mode multidrop terminal.	1-63	63	
ZCNBTP.16	/C4LCBS	Maximum number of lines that can be configured.	1-254	32†(65K) 80†(81K) 80†(96K)	Central memory size must be able to accommodate number of lines specified.  Must be greater than or equal to number of lines defined in INTER-COM MUX subtable.

† Parameter to be set at build time, depending on memory size, in deck PL99AI2 or PL99AVI for the 2550. The number of lines is set by the host at load time. These values are the default values specified in the installation decks.

## TYPE 2 STATEMENTS

The following Update DEFINE names are used during the Update that produces the SCF input to the CCI compile file. The DEFINEs select CCI software modules. These DEFINE directives must be specified for compile file generation of both CCI source and MPEDIT directives. This is controlled by extraction of the appropriate installation deck from the installation decks oldpl with an =DEFINE of the desired value.

<u>Name</u>	<u>Description</u>	<u>Acceptable Value</u>	<u>Installation Deck Defined Values</u>
MODE4	If defined, build in the mode 4 terminal interface program (TIP).	MODE4 or omitted	MODE4 if no other TIP defined
HASPTIP	If defined, build in the HASP TIP.	HASPTIP or omitted	Omitted
TIP780	If defined, build in the IBM 2780/3780 (BISYNC) TIP.	TIP780 or omitted	Omitted
ASYNCR	If defined, build in the MODE3 (TTY) TIP.	ASYNCR or omitted	ASYNCR if no other TIP defined
HASPSB0	If defined, a HASP batch output stream stopped condition is ignored. If omitted, the condition causes a disconnect after 30 seconds.	HASPSB0 or omitted	Omitted
Core size	Specifies central memory size.	65K; 81K; 96K	65K if no other memory size defined
PRU size	Specifies batch size sent to host in multiples of 640 characters.	PRU1 PRU2 PRU3	PRU2 if no other PRU size defined
STATS	If defined, statistics are dispatched normally. If omitted, statistics are discarded.	STATS or omitted	Omitted
BANNUM	Specifies number of banner pages on print files.	BAN0 BAN1 BAN2	BAN2 if no other banners defined

## TYPE 3 STATEMENTS

The system variant is set up as follows.

```
=D SCFVAR.4
*ENT, LKCYC, $xxxx.
```

xxxx is the user-supplied variant identification.

## CONFIGURATION AIDS

The following lists the 255x memory space required for available software.

<u>Feature</u>	<u>Words (decimal)</u>	<u>Area That Can Be Paged</u>
Basic software (required)	35 800	3 300
MODE4 (mode 4 TIP)	6 200	4 060
HASPTIP (HASP TIP)	6 100	4 000
TIP780 (IBM 2780/3780 BISYNC TIP)	4 850	3 300
ASYNCR (TTY, mode 3 TIP)	1 500	1 240

The space left over is used for line tables, terminal tables, and dynamic buffer allocation. Guidelines for the utilization of this space are 48 words per line, 75 words per interactive device, and 750 words per batch device.

One interactive device is defined as either a CRT display/keyboard or a TTY keyboard/printer/paper tape or equivalent.

One batch device is defined as any one of the following or its equivalent.

- Line printer
- Card reader
- Card punch

The preceding figures are approximations only; exact memory utilization is a function of block size, line speed, and so on.

### **NOTE**

Configuring a 255x for more than it can accommodate results in serious degradation of throughput. The 255x configuration is specified via both CCI 3 and INTERCOM installation parameters.

## INSTALLATION PROCEDURES

Create CDC CYBER Cross System permanent files by running either PL50I or PL50C.

### 2550-2 or 2551-1 INSTALLATION (PL99A)

Execute deck PL99AI1 to do the following tasks.

- Update the CCI source program file and the system creation file.
- Create the permanent file environment for subsequent decks.
- Create the 0D0 and 0DZ modules.
- EDITLIB 0D0 and 0DZ into the running system if ULIB is not defined.
- Create a new PL99A output tape.

Execution of PL99AI2 can begin after the CCI source program file and system creation file have been updated (after the input tape for deck PL99AI1 is unloaded).

Run deck PL99AI2, selecting the desired combination of type 1, 2, and 3 variables described earlier. This deck creates the downline load modules 0Dy and writes the output tape PL99B. As soon as the MPPPULGO and PL99AI2LIST files are cataloged, subsequent variants can be built using the variant deck PL99AVI. The downline module is EDITLIBed into the running system if ULIB is not defined.

Deck PL99AVI can be run as many times as desired to create additional load module variants from the MPPPULGO file created by PL99AI2. This deck creates the downline load module 0Dy and writes output tape PL99B. The MPLINK and MPEDIT listings are always printed. Type 1, 2, and 3 variables apply to deck PL99AVI and should be different from variables used when running deck PL99AI2.

Deck PL99AI3 acts as a cleanup deck that purges files.

#### INSTALLATION DECK PL99E

Deck PL99E uses the output tapes created by other PL99 jobs to EDITLIB load modules into the running system.

#### CORRECTIVE CODE

Corrective code releases for CCI 3 require generation of new load modules. The method used to incorporate the CCI update depends on whether it involves changes to MPEDIT initialization directives only or to the CCI source program library as well.

If the corrective code involves changes to the CCI source program library, a complete CCI build is required. If only MPEDIT directives are affected, a new macro load module may optionally be created either by a variant build (using deck PL99AVI) or by patching. (Patching involves running the MPEDIT program using the ZAPMP (changed to ABSOLMP for MPEDIT input) and SYMTAB files as input to MPEDIT and using the new MPEDIT directives.)

Example of patching the 2550 macro load module.

```
job statement.
REQUEST(MPPPU,*PF)
REQUEST(ZAPMP,*PF)
ATTACH(ABSOLMP,zapmp-file-name,ID=id-name)
ATTACH(SYMTAB,symbol-table-file-name,ID=id-name)
MPEDIT(CSET=64)
CATALOG(MPPPU,mpppu-file-name,ID=id-name)
CATALOG(ZAPMP,zapmp-file-name,ID=id-name)           Optional
end-of-record
CONST
  /NAM$ = 0Dx, where x can be 1 through F;
  set constants, if any, to be used in assignment section†
BEGIN assignment section
  reset variables to be modified†
  change memory contents†
END.
end-of-file
```

#### VERIFICATION PROGRAMS

The verification of CCI can be divided into the verification of system generation and the verification of the on-line system.

##### SYSTEM GENERATION

CCI must complete the system generation procedures without error to ensure proper operation of the 255x system. Each system building phase must finish processing with no errors before the next procedure is initiated. The NOS/BE 1 programs which can detect errors during system generation include Update, MACRO Assembler, PASCAL Compiler, Library Maintenance, Link Editor, and Post Edit Program. The following reference manuals should be consulted for the identification and explanation of specific types of errors.

Update Reference Manual

CDC CYBER Cross System 1 Macro Assembler Reference Manual

CDC CYBER Cross System 1 PASCAL Compiler Reference Manual

CDC CYBER Cross System Version 1 Link Editor and Library Maintenance Programs Reference Manual

As released, CCI should complete system generation without errors. If the installation parameters are modified with care and the restrictions on them adhered to, errors should not occur while building CCI.

##### ON-LINE SYSTEM

Refer to the INTERCOM verification procedure.

† Refer to the CDC CYBER Cross System Version 1 Link Editor and Library Maintenance Programs Reference Manual.

An example of patching the 2552 Phase 2 and Phase 1 load modules follows.

```
job statement.
REQUEST(MPPPU,*PF)
REQUEST(ZAPMP,*PF)
ATTACH(MPEDIT,ID=id-name)
ATTACH(SYM1,phase 1-symbol-table-file-name,ID=id-name)
ATTACH(SYM2,phase 2-symbol-table-file-name,ID=id-name)
ATTACH(ABSOLMP,phase 2-zapmp-file-name,ID=id-name)
COPYBR(SYM1,SYMTAB)
COPYBF(SYM2,SYMTAB)
REWIND(SYMTAB)
MPEDIT(CSET=64)
CATALOG(MPPPU,phase 2-mpppu-file-name,ID=id-name)
CATALOG(ZAPMP,zapmp-file-name,ID=id-name)
RETURN(SYMTAB,ZAPMP,MPPPU)
REWIND(SYM1,SYM2)
REQUEST(MPPPU,*PF)
REQUEST(ZAPMP,*PF)
ATTACH(ABSOLMP,phase 1-zapmp-file-name,ID=id-name)
COPYBR(SYM2,SYMTAB)
COPYBF(SYM1,SYMTAB)
REWIND(SYMTAB)
MPEDIT(CSET=64)
CATALOG(MPPPU,phase 1-mpppu-file-name,ID=id-name)
CATALOG(ZAPMP,phase 1-zapmp-file-name,ID=id-name)
end-of-record
CONST
  /NAM$ = 0Ex, where x can be 1 through F;
  set constants to be used in assignment section †
BEGIN phase 2 assignment section
  reset phase 2 variables to be modified †
  change phase 2 memory contents †
END.
end-of-record
CONST
  /NAM$ = 0E0;
  set constants to be used in assignment section †
BEGIN phase 1 assignment section
  reset phase 1 variables to be modified †
  change phase 1 memory contents †
END.
end-of-file
```

Optional

Optional

## VERIFICATION PROGRAMS

The verification of CCI can be divided into the verification of system generation and the verification of the on-line system.

### SYSTEM GENERATION

CCI must complete the system generation procedures without error to ensure proper operation of the 255x system. Each system building phase must finish processing with no errors before the next procedure is initiated. The NOS/BE 1 programs, which can detect errors during system generation, include UPDATE, MACRO Assembler, PASCAL Compiler, Library Maintenance, Link Editor, and Post Edit Program. The following reference manuals should be consulted for the identification and explanation of specific types of errors.

UPDATE Reference Manual

CDC CYBER Cross System 1 Macro Assembler Reference Manual

CDC CYBER Cross System 1 PASCAL Compiler Reference Manual

CDC CYBER Cross System Version 1 Link Editor and Library Maintenance Programs Reference Manual

As released, CCI should complete system generation without errors. If the installation parameters are modified with care and the restrictions on them adhered to, errors should not occur while building CCI.

### ON-LINE SYSTEM

Refer to the INTERCOM verification procedure.

† Refer to the CDC CYBER Cross System Version 1 Link Editor and Library Maintenance Programs Reference Manual.



### HARDWARE CONFIGURATION

CDCS 2 runs under NOS/BE and requires the same minimum hardware configuration as NOS/BE. The system control point job containing most of CDCS typically uses a field length between 100K and 134K octal words. To the user job, CDCS typically adds 1.5K octal words. AAM is not loaded at the user's control point if all AAM files are database files.

### RELEASE MATERIALS

CDCS 2 is released on the program library tape PL74. The structure of the release tape is as follows.

File 1	CDCS program library, including utilities
File 2	CDCS absolute binary
File 3	DBMSTRD (master directory utility) absolute binary
File 4	DBQRFA (quick recovery file applier utility) absolute binary
File 5	DBRCN/DBRST (reconstruct and restore utilities) absolute binary
File 6	DBQRFI (quick recovery file initialization utility) absolute binary
File 7	CDCSBTF (batch test facility) absolute binary
File 8	CDCS/DBU complete relocatable binary
File 9	CDCS object-time routines relocatable binary

### INSTALLATION REQUIREMENTS

CDCS 2 and the database utilities require BAM 1.5 and AAM 2 Extended to be installed. CDCS 2 supports only AAM 2 Extended files.

Installation of the master directory utility requires that DDL 3 directory access routines be installed.

To activate the interface between CDCS 2 and COBOL 5, refer to the installation procedure for COBOL 5. CDCS 2 does not support COBOL 4.

Installation of the utilities DBRCN/DBRST (reconstruct and restore) requires that Sort/Merge and the FORTRAN interface to Sort/Merge be installed.

CDCS 2 requires DDL 3. DDL 3 is released on PL77. The installation deck PL74I expects to find the syntax table generator SYNGEN on PL77. A debug trace of CDCS activity can be obtained by using the E option on the SYMPL compilation. Refer to the CDCS Internal Maintenance Specifications for details.

To activate system control point code in CMR, the CMR configuration parameter N.SBSYS must be set to a value greater than or equal to 2. This parameter defines the maximum number of subsystems; its default is zero.

### INSTALLATION PROCEDURE

Installation decks PL74I, PL74E, PL74O, and PL74V can be obtained from the Installation Deck program library using the procedure described in part I, section 1.

Deck PL74I serves as a program library maintenance deck in that it allows regeneration of the CDCS program library and binary files. Deck PL74E uses EDITLIB to enter CDCS into the running system or user libraries, either from the release tape or from a tape created by deck PL74I. PL74O uses the output tape created by PL74I to recreate the absolute binaries.

Because CDCS operates at a different control point from the user job, the EXIT and DMP instructions in PL74V are required for maintenance and PSR submittal. In addition, the MAP,ON directive is required in the installation deck to obtain a load map to go with the dump.

### VERIFICATION PROGRAMS

The CDCS 2 verification job, PL74V, builds all files and procedures necessary to execute a CDCS job. Operator actions are required at several points. Instructions are provided at these points by comments on PAUSE statements. Failure to set N.SBSYS as noted in the Installation Requirements causes failure of this job.





---

### RELEASE MATERIALS

DDL is released on the release tape known as PL77. The structure of the release tape is as follows.

File 1	Program library (including SYNGEN)
File 2	DDL binary; absolute format
File 3	Directory access routines; relocatable format
File 4	CDCS conversion routines; relocatable format
File 5	DDL binary; relocatable format

### HARDWARE CONFIGURATION

DDL requires the same minimum hardware configuration as NOS/BE. A minimum of 50K CM is required to execute DDL.

### INSTALLATION PROCEDURE

Installation job decks PL77I, PL77E, PL77O, and PL77V can be obtained from the installation deck PL using the procedure outlined in part I, section 1.

PL77I can be used to modify the PL, build a NEWPL, assemble and compile the entire DDL PL, and generate and save the relocatable and absolute binaries on the NEWPL. In non-ULIB mode, PL77I executes an EDITLIB to the running system DMSLIB library to make the directory access routines and conversion routines available for products that use the CDCS interface.

PL77E is used to install DDL into the running system or user libraries.

Job PL77O, applicable only in the user library method of installation, allows regeneration and replacement of absolute overlays in the user library plus creation of a new PL77 tape.

The main overlay of DDL is placed in the library NUCLEUS, the other overlays in the library SYSOVL. The directory access routines and the CDCS conversion routines are placed on the library DMSLIB.

SYNGEN, a special syntax table generator, is needed to compile DDL. SYNGEN, now on PL77, is designed to facilitate the implementation of syntax driven software. It is required to compile DDL and QU3.

DDL 3.0 can be installed in the same library as DDL 2.x.



HARDWARE CONFIGURATION

CID 1 executes in the same minimum configuration required for INTERCOM under NOS/BE.

RELEASE MATERIALS

CID is released on program library tape PL82, which contains the following files.

File 1	Source code in Update program library format
File 2	Relocatable binaries
File 3	Absolute overlays

INSTALLATION PARAMETERS

The following symbols in CID can be changed to adjust table sizes. The variable n must be a positive value. No upper limit exists, but the size of CID increases with the value of n. The symbol  $\equiv$  represents a 0-8-6 punch in 026 keypunch format.

Breakpoint table; released value is 16 breakpoints.

```
*DELETE BREAKD.8
DEF BREAKTABSIZ  $\equiv$  n $\equiv$ ;
*DELETE BREAKZ.11
TABSIZ EQU n
```

Group table; released value is 16 groups.

```
*DELETE GROUPTABSIZ  $\equiv$  n $\equiv$ ;
*DELETE GROUPZ.11
TABSIZ EQU n
```

Trap table; released value is 16 traps.

```
*DELETE TRAPTABSIZ  $\equiv$  n $\equiv$ ;
DEF TRAPXSIZ  $\equiv$  m $\equiv$ ;           m=n+3
*DELETE TRAPZ.11,12
TABSIZ EQU n
XSIZ EQU m
```

The following parameter determines the size that the 54-table of program can become before requiring CID to recreate its overlays at debug time. The release value is 10<sub>8</sub> words of extra 54-table information.

```
*DELETE DBUGI.85
ROOM54 EQU n
```

INSTALLATION PROCEDURES

The IPARAM micro OS.NAME must be SCOPE and the Update directive \*DEFINE NOSBE must be included as shown in installation job PL82L.



---

### RELEASE MATERIALS

The PL/I compiler and run-time system reside on release tape PL79. The structure of PL79 is as follows.

File 1	Update program library of the compiler and run-time system
File 2	Relocatable binary records of the compiler, including the common code generator
File 3	Relocatable binary records of the run-time system
File 4	Absolute binary records of the compiler, including the common code generator

If a compiler is to be built from the program library, the common code generator's program library must also be used. It resides on file 1 of release tape PL83. COMPCOM and the common common decks must be obtained from the COMPASS program library on PL2.

### REQUIREMENTS

To ensure correct code generation, the MODEL micro in IPARAMS must be set to the proper value for the host machine.

Because PL/I uses Update (with the multiple OLDPL feature), COMPASS, FORTRAN Extended 4, SYMPL, CYBER Loader, and CYBER Record Manager for installation, it must follow these products in the build sequence.

The common code generator (release tape PL83) must be used as the compiler code generator.

Proper execution of programs compiled under PL/I requires the FORTRAN Common Library 4 (both math and I/O), BAMLIB, and AAMLIB.

Installation of PL/I requires a field length of 105000<sub>8</sub> words.

### INSTALLATION OPTIONS

The PL/I program library, PL79, is distributed with installation parameters set properly for normal installations. The system text IPTEXT should contain parameter values consistent with the CYBER model on which the compiler is generated and executed. PLITEXT, CMPLTXX, RTSTEXT, and CCGTEXT select symbol definitions from IPTEXT for use by the PL/I compiler, code generator, and run-time system. No other references to these symbols exist in the compiler or run-time system.

The compiler installation options are located in the common deck OPTIONS and the deck PLI. OPTIONS is called by PLITEXT, CMPLTXX, and CCGTEXT. Because these texts are used globally, the compiler should be reinstalled whenever parameters in OPTIONS are changed. Installation parameters in PLI can be revised through a standard maintenance run (installation deck PL79I1). This job performs an UPDATE,N. against the PL79 output tape, and a COPYL against the relocatable binary from the tape. If only a few routines require changing, this job is much faster than PL79I.

The run-time system installation options are located in RTSTEXT. The run-time system should be reinstalled when any of its options are changed.

Current Update sequence numbers can be obtained by assembling PLITEXT, PLI, and RTSTEXT.

## PROGRAM LIBRARY STRUCTURE

When a full Update is performed on PL79, the following records are written on the compile file.

	<u>Overlay</u>	<u>Deck Names</u>
1. Skeleton of compiler binaries		PLISKEL
2. Assembly text for the compiler		PLITEXT
3. Assembly text unique to pass three		P3TEXT
4. Primary main overlay	(0,0)	PLI
Secondary main overlay	(1,0)	PLI10 - SNAP
Passes one and two	(1,1)	PLI11 - PLMACH2
5. Pass three	(1,3)	PLI13 - P3MOD6
6. Pass four COMPASS	(1,4)	PLI14 - MTABLE
Pass five COMPASS	(1,5)	PLI15 - PLEPT
Code generator bridge	(1,6)	PLI16 - MACROX
7. Table manager SYMPL	(1,0)	TSPREAD - TMSPACE
Pass four SYMPL	(1,4)	P4PRO - P4SUB
Pass five SYMPL	(1,5)	SREF3 - PLEEND
8. Run-time assembly text		RTSTEXT
9. Run-time record I/O		PLIO - PLEVIO
10. Run-time stream I/O		STRIO - PLABTO
11. Run-time built-in functions		ABS=C - VERIFY
12. Run-time on-units		AREA= - ZDIV=
13. Run-time control, string handling, and conversion		PLALBK - PLMOVW

## INSTALLATION DECKS

The following installation decks are supplied.

PL79I	Performs a full Update and assembly/compilation of the compiler and run-time system.
PL791I	Performs a partial Update and assembly/compilation of the compiler and run-time system. This deck is provided as an alternative to the execution of PL791. It is of value when CRM or CMM interfaces have not changed and decks are correctly ordered on the Update compile file.
PL79E	Performs an EDITLIB of the compiler and run-time system from a release format tape into the host system or user libraries.
PL79V	Verifies PL/I installation
PL79O	Reformats the absolute binaries of the compiler and run-time system, and produces a tape that can be entered into the running system or user libraries through PL79E.

These decks can be obtained using the information supplied in part I, section 1.

---

RELEASE MATERIALS

FCL 4 is released on one reel of tape (PL8) with the following structure.

File 1	Program library of FCL 4 math and I/O routines in Update format
File 2	Relocatable binaries of FCL 4 routines
File 3	Absolute binary of PMD

INSTALLATION OPTIONS

MATHTXT and FCLTEXT select installation options from IPTEXT for use by FCL 4. No other direct references to IPARAMS exist in the product.

PROGRAM LIBRARY STRUCTURE

When a full Update is performed, the following records are written on the compile file.

- FCLTEXT Text used to assemble nonmath routines
- Nonmath relocatable routines I/O, Debug, Sort/Merge interface, and miscellaneous routines
- PMD Post mortem dump relocatable routines
- Miscellaneous encapsulated routines Loader directives and routines to be encapsulated
- MATHTXT Text used to assemble math routines
- Math routines Call-by-name and call-by-value mathematical routines

INSTALLATION DECKS

The following installation decks are supplied.

- |       |                                                                                                                                                                                                               |
|-------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| PL8I1 | Performs a full Update and an assembly of the math routines.                                                                                                                                                  |
| PL8I2 | Performs an assembly of the I/O and PMD routines and writes a release format tape.                                                                                                                            |
| PL8E  | Performs an EDITLIB of FCL 4 math, I/O, and PMD routines from a release format tape into the running system or user libraries.                                                                                |
| PL8E1 | Performs a running system EDITLIB of FCL 4 math routines from the file cataloged by PL8I1 such that subsequent installation of SYMPL (PL6A1) is possible. This deck is not applicable to user library builds. |





---

### RELEASE MATERIALS

FDBF is released on the release tape known as PL66. The structure of the release tape is as follows.

File 1	Program library
File 2	DDLDF binary; absolute format
File 3	DML binary; absolute format
File 4	DDLDF binary; relocatable format
File 5	DML binary; relocatable format
File 6	Object time routines binary

### HARDWARE CONFIGURATION

FDBF requires the same minimum configuration as NOS/BE. A minimum of 60K CM is required to execute DDLDF (the FORTRAN Extended 4 subschema compiler) and 45K CM to execute DML (the data manipulation language preprocessor).

### NOTES AND CAUTIONS

FDBF requires installation of DDL 3. The installation deck expects to find the syntax table generator, SYNGEN, on PL77.

### INSTALLATION PROCEDURE

Installation job decks PL66I, PL66E, PL66O, and PL66V can be obtained from the installation deck PL using the procedure outlined in part I, section 1.

PL66I does a full UPDATE, compilation, and assembly of DDLDF, DML, and the object routines. It produces a new tape with the same structure as the release tape, and optionally, either creates or updates user libraries.

PL66E installs or replaces DDLDF, DML, and the object routines on the running system or user libraries with those on the release tape or the tape created by PL66I or PL66O.

PL66O allows regeneration of absolute overlays. This job produces the same tape and libraries as job PL66I.

The use of the LOCLIB parameter in the installation decks causes PL66I, PL66E, and PL66O to editlib FDBF binaries into a local library instead of USERNUC/USEROV/UDMSL or the running system. This option provides users with a more flexible approach to installation.

The main overlay of DDLDF resides in NUCLEUS, and the other overlays reside in SYSOVL. DML resides in NUCLEUS. The object routines reside in DMSLIB.

### VERIFICATION PROGRAM

Job PL66V can be run to verify the correct installation of FDBF.



---

### RELEASE MATERIALS

ALGOL Version 5 resides on release tape PL72. The structure of the release tape is as follows.

File 1	Program library in UPDATE format
File 2	Compiler relocatable binary
File 3	Compiler absolute binary
File 4	Relocatable binary of object routines and absolute overlay of symbol table
File 5	Absolute overlay of user system text for COMPASS interface

### LIMITATIONS

The ALGOL Version 5 compiler can be executed from any file or user library provided one of the following conditions is true.

- The main overlay and all other overlays except the symbol table reside on the same file or library.
- The main overlay resides in NUCLEUS and all other overlays except the symbol table reside in SYSOVL.

The object library, which should include the symbol table overlay, should be named ALG5LIB.

All files used by the compiler are formatted according to applicable operating system standards. The formats of these files cannot be changed using FILE control statements.

### INSTALLATION PARAMETERS

The micro MODLEVEL is used to incorporate the modification level in the object code and source listing; the level can be changed through the ML parameter on the COMPASS control statement.

### INSTALLATION PROCEDURES

Installation job decks PL72I, PL72E, PL72O, and PL72V can be obtained from the installation deck program library using the procedure outlined in part I, section 1.

Job PL72I uses the release tape as input to create a tape of the same structure, containing an updated version of the PL and newly generated binaries. It references CPUTEXT, IPTEXT, and IOTEXT.

Job PL72E uses the release tape or the tape generated by PL72I as input to EDITLIB to enter all binaries into the running system.

Job PL72O reformats the absolute compiler overlays from the relocatable binaries contained on the release tape and generates a new tape with the same structure.

If the user library method has been selected for job PL72I, job PL72E need not be run.

### VERIFICATION PROGRAM

Job PL72V can be run to verify the correct installation of ALGOL Version 5.



---

**RELEASE MATERIALS**

The FORTRAN 4/5 Conversion Aid resides on release tape PL65. The structure of the release tape is as follows.

File 1	FORTRAN 4/5 Conversion Aid source in UPDATE program library format
File 2	Conversion Aid absolute binary
File 3	Conversion Aid relocatable binary

**HARDWARE REQUIREMENTS**

The FORTRAN 4/5 Conversion Aid can be maintained on the same hardware configuration as that required for FORTRAN Extended 4.

**INSTALLATION PROCEDURES**

Installation job decks PL65I, PL65E, and PL65V can be obtained from the installation deck PL using the procedure outlined in part I, section 1.

PL65I uses the release tape as input to generate a new PL65 tape containing a revised program library, absolute binary, and relocatable binary file.

PL65E uses the release tape or the tape generated by PL65I to enter FORTRAN 4/5 Conversion Aid into the running system.

**VERIFICATION PROGRAM**

PL65V can be run to verify the correct installation of the FORTRAN 4/5 Conversion Aid.



---

### RELEASE MATERIALS

FORTRAN 5 is released on one reel of tape (PL63) which contains the compiler. The installation of FCL 5 (PL64) mathematical and I/O libraries is required for FORTRAN 5 execution (refer to section 39).

The structure of PL63 is as follows.

File 1	Program library of the FORTRAN 5 compiler
File 2	Relocatable binary
File 3	Relocatable binary
File 4	Absolute overlay binary

### LIMITATIONS

Because all code generated by the compiler assumes the existence of the Integer Multiply hardware option, all applicable Integer Multiply FCOs must be installed.

If FORTRAN 5 is installed on a CYBER 70 model 71, 72, or 73, or a CYBER 170 model 171, 172, 173, 174, 720, or 730 with the MODEL installation parameter (in IPARAMS) correspondingly set, the object code produced will execute properly but will not be optimal for a model 74, 175, 750, or 760. If MODEL is set to 74 or 175, the object code produced will execute properly on a model 71, 72, 73, 74, 171, 172, 173, 174, 175, 720, 730, 750, or 760, but will be optimal only for the model selected. If the MODEL parameter is set to 176, the compiled object code will not execute correctly on other models when the source programs contain LEVEL 2 (direct access LCM) statements, but will execute correctly although not optimally on other models when the source programs do not contain LEVEL 2 statements.

Most user programs written in FORTRAN Extended 4 will require translation before they compile properly under FORTRAN 5. Refer to section 37 for installation instructions and to the FORTRAN 4/5 Conversion Aid Reference Manual for a product description.

### INSTALLATION PARAMETERS

Installation parameters can be obtained by assembling FTN5TXT and/or FTN, depending on the parameters of interest. FTN contains the installation parameters for default control statement settings, control statement error processing, default file names, input/output buffer length, and compiler overlay library names. The remaining parameters are in OPTIONS (called by FTN5TXT).

The compiler and CCG should be reinstalled whenever parameters in OPTIONS are changed. Installation parameters in COMFCIP (called by decks FTN and INIT00) can be revised through a standard maintenance run (installation deck PL63I) if both FTN and INIT00 are reassembled.

### INSTALLATION PROCEDURE

Compiler installation job decks PL63I and PL63E, and verification program PL63V can be obtained from the installation deck program library using the procedure outlined in part I, section 1. PL63I updates the program library, producing a new program library tape including supplemental binary files. Deck PL63E must be run following PL63I but before attempting installation of the object library when the running system modification approach to building systems is used.

Deck PL63I references IPTEXT and CPUTEXT; part III of this document contains a cross reference map of referencing routines versus IPARAMS symbols. Deck PL63I also requires access to the COMPASS program library to acquire the common deck COMPCOM and the common common decks, and to the CCG program library to acquire the common code generator.

Decks PL63E and DST3 need not be run if the user library approach is being followed.





---

### RELEASE MATERIALS

FCL 5 is released on one reel of tape (PL64) with the following structure:

File 1	Program library of FCL 5 math and I/O routines in Update format
File 2	Relocatable binaries of FCL 5 routines
File 3	Absolute binary of PMD

### INSTALLATION OPTIONS

MATHTXT and FCLTEXT select installation options from IPTEXT for use by FCL 5. No other direct references to IPARAMS exist in the product.

### PROGRAM LIBRARY STRUCTURE

When a full Update is performed, the following records are written to the compile file:

- FCLTEXT                      Text used to assemble nonmath routines.
- Nonmath relocatable routines      I/O, character routines, Sort/Merge interface, and miscellaneous routines.
- PMD                              Post mortem dump relocatable routines.
- Miscellaneous encapsulated routines      Loader directive and routines to be encapsulated.
- MATHTXT                      Text used to assemble math routines.
- Math routines                  Call-by-name and call-by-value mathematical routines.

### INSTALLATION PROCEDURES

The following installation decks are supplied.

PL64I1	Performs a full Update and an assembly of the math routines.
PL64I2	Performs an assembly of the I/O and PMD routines and writes a release format tape.
PL64E	Performs an EDITLIB of FCL 5 math, I/O, and PMD routines from a release format tape into the running system or user libraries.



---

### RELEASE MATERIALS

EXPORT High Speed (HS) is released on one reel of magnetic tape (PL80) containing the EXPORT HS program library in Update format as file one.

### HARDWARE CONFIGURATION

In addition to the minimum configuration required for NOS/BE, EXPORT HS requires the following:

#### At the central site

- 1 6673 or 6674 multiplexer
- 1 dedicated peripheral processor and channel
- 1 control point with 4300g to about 65100g CM words, depending upon terminal activity and hardware configuration
- 1 301B or 303 DATAPHONE† Data Set or CDC 358-3 transceiver

#### At the remote site

- 1 CDC 1700 remote terminal system
- 1 301B or 303 DATAPHONE Data Set or CDC 358-3 transceiver
- 1 1747 Data Set controller or 774-2 IGS console

The model of the data set or transceiver at the remote site must match that of the central site. An additional 6673 or 6674 multiplexer on a second dedicated channel and peripheral processor can be added and run at the same control point.

### LIMITATIONS

When 40.8KB communication lines are used, each peripheral processor can service up to four terminals. However, when 50KB lines are used, each peripheral processor can service a maximum of only three terminals. This is a multiplexer hardware limitation.

EXPORT HS can communicate with several different IMPORT packages having similar, but not identical, commands. Informative messages or error diagnostics issued by the various IMPORT packages may differ in minor respects.

Interactive or graphics data streams are not supported. If the 1700 terminal being used has a 274/774 graphics display, any attempt to use the display or keyboard causes communications to be terminated until the IMPORT terminal is reloaded.

### INSTALLATION PARAMETERS

The following symbols in deck 1HS can be changed by the installation.

- |       |                                                                                                                                                                                                                                                                                                           |
|-------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| COPY  | Used to define the number of EXPORT drivers. If COPY is set to EXP, only one 6673/6674 is to be used. If set to EXP1, two 6673/6674s are assumed and 1HS is used to drive the first. If COPY is set to EXP2, 1HS must be renamed to 2HS and used to drive the second 6673/6674. The release value is EXP. |
| CMBL  | Length of the central memory I/O buffer for each data stream. These buffers are allocated only as necessary and must be at least 101g words long. The release value is 1001g.                                                                                                                             |
| DBLEN | Default transmission buffer length in 12-bit bytes. This value must correspond to the value defined in the IMPORT system. The release value is 245g.                                                                                                                                                      |

---

† DATAPHONE is a trademark of the American Telephone and Telegraph Company.

## EQUIPMENT STATUS TABLE

The EST, established when deck PL1AI is run to install NOS/BE, must contain an entry for each multiplexer used by EXPORT. The EST macro has the following format.

SC        EST            parameters            (of the form key=value)

Macro parameters used by EXPORT HS include the following.

CH=      Channel for 6673/6674 multiplexer  
EQP=      Equipment number for multiplexer  
MOD=      OFF if off, otherwise do not use  
MUX=      Index to multiplexer table

Refer to part II, section 1, Equipment Configuration, and section 28, EST Entry, for more detail.

## CMR MULTIPLEXER TABLE

The CM resident multiplexer table is shared by EXPORT HS and INTERCOM 5 to provide data on the hardware configuration of the installation and to record parameters. The multiplexer table entry can be defined as follows.

label     MUX73    n  
label     MUX74    n

label     Value used in the MUX=value parameter in the EST macro.  
n         Number of ports to be serviced: maximum of 2 for 6673, 4 for 6674.

Refer to part II, section 28, CMR Multiplexer Table, for more detail.

## INSTALLATION PROCEDURES

Install EXPORT HS from PL80 with decks PL80I and PL80E.

Assemble CMR with the proper EST and multiplexer table entries. Generate a deadstart tape with the new CMR and EXPORT HS programs.

---

RELEASE MATERIALS

Data Catalogue 2 is released on one reel of tape (PL73) with the following structure.

File 1	Program library of Data Catalogue 2
File 2	Absolute binary of DCUPD
File 3	Absolute binary of DCSEL
File 4	Absolute binary of DCRPT
File 5	Absolute binary of DCRET
File 6	Absolute binary of DCCONVT
File 7	Absolute binary of DCUTL
File 8	Absolute binary of DCIDX
File 9	Absolute binary of DCCONGN
File 10	Relocatable binary of DCUPD
File 11	Relocatable binary of DCSEL
File 12	Relocatable binary of DCRPT
File 13	Relocatable binary of DCRET
File 14	Relocatable binary of DCCONVT
File 15	Relocatable binary of DCUTL
File 16	Relocatable binary of DCIDX
File 17	Relocatable binary of DCCONGN

INSTALLATION PROCEDURE

Data Catalogue 2 requires installation of the COBOL 5 compiler and library.

Data Catalogue 2 cannot be added to the running system. The product must run from permanent files.

Job decks PL73I and PL73C can be obtained from the installation deck program library using the procedure outlined in part I, section 1.

The installation jobs function as follows:

PL73I	Updates the program library with modifications to produce a new program library tape including relocatable and absolute binary files.
PL73C	Catalogs the Data Catalogue 2 binaries from the tape created by PL73I or the released tape as public files. The user is responsible for introducing the installation-defined password required to catalog files under ID=PUBLIC.

Because the PL73 installation jobs do not enter program binaries into libraries, the DSTn jobs are not applicable.



## **PART III**

Cross-reference listing showing routines that reference installation parameters.





\*\*\*\*PL1A\*\*\*\*

\*\*\*\*\*

IPPARAMS

\*\*\*\*\*

IP.ACNT	1AJ					
IP.BOVF	BBJ	1IB				
IP.BRCL	CPSCH					
IP.CMBL	REGEBUF	GETRAND	STORE	FLUSHST		
IP.CMPID	CMR					
IP.BCFAP	CLRCCEM	ECSUB	CEM			
IP.CMU	CED					
IP.CPLM	7AJ	RPV				
IP.C63	1RN	1MT	1P2	1P4	1P3	1P1
	1WI	1RV	1NO	1N2	1N3	1R2
	1R3	1TF	1WS	1NW	8T3	2TB
	1CR	1CT	1CS	1RT	1RS	1NR
	1W9	1C9	1R9	4LB	4LC	
IP.CSET	CED	1RN	1IT	2IR	3II	2IW
	2IA	2IC	2ID	2IP	2IT	2IX
	1MT	1P2	1P4	1P3	1P1	1WI
	1RV	1NO	1N2	1N3	1R2	1R3
	1TF	1WS	1NW	8T3	2TB	1CR
	1CT	1CS	1RT	1RS	1NR	1W9
	1C9	1R9	4LB	7T1	7T2	4LC
	1LC					
IP.C176	CED	CMR	CPMTR	SEGLINK	MTR	
	DSD	1SC				
IP.DCT	CMR					
IP.DECR	CCP	CSWP				
IP.ECFL	CMR					
IP.DBAL	1IB	1AB				
IP.ECSB	CMRTEXT	CED	IRCP	STL	CMR	CPMTR
	EXBOOT	MTR	DSD	1AJ		
	7EC	1EJ	3DO	1RN	1SP	MEM
	REQ	3RQ	QAF	QAC	1IB	1SI
	1SO	CEM	TDS			
IP.FTHRL	CMR					
IP.IACES	1AJ					
IP.ILCMD	DSD					
IP.INT5	PPTEXT	SCPTTEXT	CMR	DSD		
IP.IOLM	7AJ	RPV				
IP.IOD	1RN					
IP.IQPW	1IB					
IP.IRCL	CPSCH					
IP.IUSID	CMR					
IP.LINK	DSD					
IP.LVF	1EJ	4EJ	1RN	1IB	1TJ	2VJ
	6RD					
IP.MCPU	CMRTEXT	CED	CMR	CPMTR	SETST	CPUST
	USERMOD	TIMSEG	SSCSEG	SSFSEG	CPSS	PACKAGE
	LINK	RESCH	UPM	CCP	SYSIDLE	WOR
	CMDATA					
	RAGET	MTR				
	DSD	7AJ	4EJ	1SP	DIS	1IB
	1SI	1SO	1TJ	2VJ	RPV	
IP.MECS	CMR	CPSS	PACKAGE	CPECSM		
	MTR	DSD	4EJ	DIS	MEM	
	1IB	1SI	1SO	1TJ	2VJ	
IP.MMS	1AJ					
IP.MPPU	CMR	SPRBMGR	MTR			
IP.MPR	4EJ	1TJ	2VJ			
IP.MSCT	1IB					
IP.MSLM	7AJ	RPV				
IP.MTL	4EJ	1TJ	2VJ			
IP.MXTIN	UPM					
IP.MXQT	CSWP					
IP.NDFS	1EJ					
IP.NJFL	BBJ					
IP.OPRI	6RD					
IP.OOD	1RN					
IP.POSFL	MTR	1SO				
IP.PPS2X	CED					
IP.RBINT	1RN					

IP.ROFL	UPM	1SO				
IP.SCHDE	IRCP	DSD	1EJ	4EJ	3RQ	3MN
	1CL	3IC	3IE	3IR	3IW	1IB
	1TJ	2VJ	1TS	1LC		
IP.SFL	DSD	1AJ	4EJ	1SI	1TJ	2VJ
	STS	1AB				
IP.SIDLE	CMR	CPMTR	RESCH	CPSCH	SYSIDLE	CMRDIR
	CMDATA	MTR				
	DSD	1SC	TDS			
IP.SIOL	4EJ	1TJ	2VJ			
IP.SLIB	CMR					
IP.SMS	1IB	1SI				
IP.SPR	4EJ	1TJ	2VJ	6RD		
IP.SPT	IRCP	CMR	CPMTR	CPSM	CPSM	CPSCH
	BBJ					
	1SO					
IP.SRMS	CED	STL	1RN			
IP.STEX	DSD					
IP.STL	4EJ	1TJ	2VJ			
IP.SYSE	CMR					
IP.TCPUB	CPMTR	TIMSEG	RESCH			
IP.TOVF	BBJ	1IB				
IP.TYPE	LINK	RESCH	CPSM	SCHRES		
IP.VER	CMR					
IP.XJ	CPMTR	CPUST	RESCH	USERR	SCHRES	SEGPAP
	RAGET					
	CACT	MTR	DSD			
IP.YMD	IRCP	DSD				
IP.819	IPTXT	CMRTEXT	CED	IRCP	STL	CMR
	CPSM	CMRDIR	MTR	DSD	CIO	
	1AJ	7EC	1EJ	3D0	1S5	1RN
	1SP	DIS	MEM	REQ	3RQ	ACE
	1RP	1CL	1TR	1T0	6WM	1MH
	1IT	2IW	2IA	2IB	2IC	2ID
	2IL	2IO	2IP	2IT	2IX	3IC
	3IE	3IF	3IJ	3IL	3IM	3IN
	3IO	3IR	3IV	3IW	1MT	SSH
	QAF	QAC	1P2	1P4	1P3	1P1
	1WI	1RV	1N0	1N2	1N3	1R2
	1R3	1TF	1WS	1NW	8T3	2TB
	1CR	1CT	1CS	1RT	1RS	1NR
	1W9	1C9	1R9	1SC	1IB	1SI
	1SO	1TJ	2VJ	4LB	4LC	6RD
	CEM	1TS	RPV	SSC	STS	TDS
	1LC	1AB	REQUEST	LABEL		
MODEL	IPTXT	CMRTEXT	CED	IRCP	STL	CMR
	MTR	DSD	CIO	1AJ	1EJ	3D0
	1S5	1RN	1SP	DIS	MEM	REQ
	3RQ	ACE	1RP	1CL	1TR	1T0
	6WM	1MH	1IT	2IW	2IA	2IB
	2IC	2ID	2IL	2IO	2IP	2IT
	2IX	3IC	3IE	3IF	3IJ	3IL
	3IM	3IN	3IO	3IR	3IV	3IW
	1MT	SSH	QAF	QAC	1P2	1P4
	1P3	1P1	1WI	1RV	1N0	1N2
	1N3	1R2	1R3	1TF	1WS	1NW
	8T3	2TB	1CR	1CT	1CS	1RT
	1RS	1NR	1W9	1C9	1R9	1SC
	1IB	1SI	1SO	1TJ	2VJ	4LB
	4LC	6RD	CEM	1TS	RPV	SSC
	STS	TDS	1LC	1AB	REQUEST	LABEL
OS.NAME	IPTXT	CMRTEXT	CED	IRCP	STL	CMR
	MTR	DSD	CIO	1AJ	1EJ	3D0
	1S5	1RN	1SP	DIS	MEM	REQ
	3RQ	ACE	1RP	1CL	1TR	1T0
	6WM	1MH	1IT	2IW	2IA	2IB
	2IC	2ID	2IL	2IO	2IP	2IT
	2IX	3IC	3IE	3IF	3IJ	3IL
	3IM	3IN	3IO	3IR	3IV	3IW
	1MT	SSH	QAF	QAC	1P2	1P4
	1P3	1P1	1WI	1RV	1N0	1N2
	1N3	1R2	1R3	1TF	1WS	1NW
	8T3	2TB	1CR	1CT	1CS	1RT

	1RS	1NR	1W9	1C9	1R9	1SC
	1IB	1SI	1S0	1TJ	2VJ	4LB
	4LC	6RD	CEM	1TS	RPV	SSC
	STS	TDS	1LC	1AB	REQUEST	LABEL
OS.VER	IPTEXT	CMRTEXT	CED	IRCP	STL	CMR
	MTR	DSD	CIO	1AJ	1EJ	300
	1S5	1RN	1SP	DIS	MEM	REQ
	3RQ	ACE	1RP	1CL	1TR	1T0
	6WM	1MH	1IT	2IW	2IA	2IB
	2IC	2ID	2IL	2IO	2IP	2IT
	2IX	3IC	3IE	3IF	3IJ	3IL
	3IM	3IN	3IO	3IR	3IV	3IW
	1MT	SSH	QAF	QAC	1P2	1P4
	1P3	1P1	1WI	1RV	1N0	1N2
	1N3	1R2	1R3	1TF	1WS	1NW
	8T3	2TB	1CR	1CT	1CS	1RT
	1RS	1NR	1W9	1C9	1R9	1SC
	1IB	1SI	1S0	1TJ	2VJ	4LB
	4LC	6RD	CEM	1TS	RPV	SSC
	STS	TDS	1LC	1AB	REQUEST	LABEL
PR.BATCH	IPTEXT	CMRTEXT	CED	IRCP	STL	CMR
	MTR	DSD	CIO	1AJ	1EJ	300
	1S5	1RN	1SP	DIS	MEM	REQ
	3RQ	ACE	1RP	1CL	1TR	1T0
	6WM	1MH	1IT	2IW	2IA	2IB
	2IC	2ID	2IL	2IO	2IP	2IT
	2IX	3IC	3IE	3IF	3IJ	3IL
	3IM	3IN	3IO	3IR	3IV	3IW
	1MT	SSH	QAF	QAC	1P2	1P4
	1P3	1P1	1WI	1RV	1N0	1N2
	1N3	1R2	1R3	1TF	1WS	1NW
	8T3	2TB	1CR	1CT	1CS	1RT
	1RS	1NR	1W9	1C9	1R9	1SC
	1IB	1SI	1S0	1TJ	2VJ	4LB
	4LC	6RD	CEM	1TS	RPV	SSC
	STS	TDS	1LC	1AB	REQUEST	LABEL
PR.NUMB	CMR					
PR.INT	IPTEXT	CMRTEXT	CED	IRCP	STL	CMR
	MTR	DSD	CIO	1AJ	1EJ	300
	1S5	1RN	1SP	DIS	MEM	REQ
	3RQ	ACE	1RP	1CL	1TR	1T0
	6WM	1MH	1IT	2IW	2IA	2IB
	2IC	2ID	2IL	2IO	2IP	2IT
	2IX	3IC	3IE	3IF	3IJ	3IL
	3IM	3IN	3IO	3IR	3IV	3IW
	1MT	SSH	QAF	QAC	1P2	1P4
	1P3	1P1	1WI	1RV	1N0	1N2
	1N3	1R2	1R3	1TF	1WS	1NW
	8T3	2TB	1CR	1CT	1CS	1RT
	1RS	1NR	1W9	1C9	1R9	1SC
	1IB	1SI	1S0	1TJ	2VJ	4LB
	4LC	6RD	CEM	1TS	RPV	SSC
	STS	TDS	1LC	1AB	REQUEST	LABEL
PR.SCP	IPTEXT	CMRTEXT	CED	IRCP	STL	CMR
	MTR	DSD	CIO	1AJ	1EJ	300
	1S5	1RN	1SP	DIS	MEM	REQ
	3RQ	ACE	1RP	1CL	1TR	1T0
	6WM	1MH	1IT	2IW	2IA	2IB
	2IC	2ID	2IL	2IO	2IP	2IT
	2IX	3IC	3IE	3IF	3IJ	3IL
	3IM	3IN	3IO	3IR	3IV	3IW
	1MT	SSH	QAF	QAC	1P2	1P4
	1P3	1P1	1WI	1RV	1N0	1N2
	1N3	1R2	1R3	1TF	1WS	1NW
	8T3	2TB	1CR	1CT	1CS	1RT
	1RS	1NR	1W9	1C9	1R9	1SC
	1IB	1SI	1S0	1TJ	2VJ	4LB
	4LC	6RD	CEM	1TS	RPV	SSC
	STS	TDS	1LC	1AB	REQUEST	LABEL
PR.SYS	CMR	CPUST				
*****						
CIOCOM						
*****						
IP.D7DN	CED					

IP.NBCD	1EJ	3MN	8T3	6LC	6L3	1TS
IP.NBRK	3MN					
IP.NDEN	CMRTEXT	CED	CMR	DSD	CIO	1EJ
	4EJ	3D0	4D0	1RN	4ES	REQ
	3RQ	3MN	1RP	1CL	2TC	1TR
	1T0	10P	1BT	6WM	1MF	1MH
	1IT	2IW	2IA	2IB	2IC	2ID
	2IL	2IO	2IP	2IT	2IX	3IC
	3IE	3IF	3IJ	3IL	3IM	3IN
	3IO	3IR	3IV	3IW	1MT	1P2
	1P4	1P3	1P1	1WI	1RV	1N0
	1N2	1N3	1R2	1R3	1TF	1WS
	1NW	8T3	2TB	1CR	1CT	1CS
	1RT	1RS	1NR	1W9	1C9	1R9
	1IB	1TJ	2VJ	4LB	4LC	1TS
	REQUEST	LABEL				
IP.NOISE	1IT	3II	1MT	1P1	1N0	1N2
	1N3	1R2	1R3	1TF	1WS	1NW
	2TB	1RS	1NR			
IP.NOIS9	1IT	3II	1P1	1N0	1N2	1N3
	1R2	1R3	1TF	1NW	2TB	1NR
IP.NTCN	1RN	1MT	1P2	1P4	1P3	1P1
	1WI	1RV	1N0	1N2	1N3	1R2
	1R3	1TF	1WS	1NW	8T3	2TB
	1CR	1CT	1CS	1RT	1RS	1NR
	1W9	1C9	1R9	4LB	4LC	
IP.PCL	1RN	1MT	1P2	1P4	1P3	1P1
	1WI	1RV	1N0	1N2	1N3	1R2
	1R3	1TF	1WS	1NW	8T3	2TB
	1CR	1CT	1CS	1RT	1RS	1NR
	1W9	1C9	1R9	4LB	4LC	
IP.PTCN	1RN	1MT	1P2	1P4	1P3	1P1
	1WI	1RV	1N0	1N2	1N3	1R2
	1R3	1TF	1WS	1NW	8T3	2TB
	1CR	1CT	1CS	1RT	1RS	1NR
	1W9	1C9	1R9	4LB	6LC	4LC
IP.RCYC	2IO	3IL	4LB	4LC	LABEL	
IP.RPE1	1RN	1MT	1P2	1P4	1P3	1P1
	1WI	1RV	1N0	1N2	1N3	1R2
	1R3	1TF	1WS	1NW	8T3	2TB
	1CR	1CT	1CS	1RT	1RS	1NR
	1W9	1C9	1R9	4LB	4LC	
IP.RPE2	1RN	1MT	1P2	1P4	1P3	1P1
	1WI	1RV	1N0	1N2	1N3	1R2
	1R3	1TF	1WS	1NW	8T3	2TB
	1CR	1CT	1CS	1RT	1RS	1NR
	1W9	1C9	1R9	4LB	4LC	
IP.TDEN	CMRTEXT	CED	CMR	DSD	CIO	1EJ
	3D0	4D0	1RN	4ES	REQ	3RQ
	3MN	1RP	1CL	2TC	1TR	1T0
	10P	1BT	6WM	1MF	1MH	1IT
	2IW	2IA	2IB	2IC	2ID	2IL
	2IO	2IP	2IT	2IX	3IC	3IE
	3IF	3IJ	3IL	3IM	3IN	3IO
	3IR	3IV	3IW	1MT	1P2	1P4
	1P3	1P1	1WI	1RV	1N0	1N2
	1N3	1R2	1R3	1TF	1WS	1NW
	8T3	2TB	1CR	1CT	1CS	1RT
	1RS	1NR	1W9	1C9	1R9	1IB
	1TJ	2VJ	4LB	4LC	1TS	REQUEST
	LABEL					
IP.TS6	CMRTEXT	CED	CMR	DSD	CIO	1EJ
	3D0	4D0	1RN	4ES	REQ	1RP
	1CL	2TC	1TR	1T0	10P	1BT
	6WM	1MF	1MH	1IT	2IW	2IA
	2IB	2IC	2ID	2IL	2IO	2IP
	2IT	2IX	3IC	3IE	3IF	3IJ
	3IL	3IM	3IN	3IO	3IR	3IV
	3IW	1MT	1P2	1P4	1P3	1P1
	1WI	1RV	1N0	1N2	1N3	1R2
	1R3	1TF	1WS	1NW	8T3	2TB
	1CR	1CT	1CS	1RT	1RS	1NR
	1W9	1C9	1R9	1IB	1TJ	2VJ
	4LB	4LC	1TS	REQUEST	LABEL	

IP.WEC ***** ECSCOM ***** IP.CMBL IP.CYSTP IP.EBUF	REQUEST	LABEL				
	REQEBUF DRVS IRCP	GETRAND	STORE	FLUSHST		
	IPARAMS	SORTED BY		ROUTINE		
ACE	IP.819 PR.SCP	MODEL	OS.NAME	OS.VER	PR.BATCH	PR.INT
BBJ	IP.80VF	IP.NJFL	IP.SPT	IP.TOVF		
CACT	IP.XJ					
CCP	IP.DECR	IP.MCPU				
CED	IP.CMU IP.SRMS PR.INT	IP.CSET IP.819 PR.SCP	IP.C176 MODEL IP.D7DN	IP.ECSB OS.NAME IP.NDEN	IP.MCPU OS.VER IP.TDEN	IP.PPS2X PR.BATCH IP.TSG
CEM	IP.BCFAP PR.BATCH	IP.ECSB PR.INT	IP.819 PR.SCP	MODEL	OS.NAME	OS.VER
CIO	IP.819 PR.SCP	MODEL IP.NDEN	OS.NAME IP.TDEN	OS.VER IP.TSG	PR.BATCH	PR.INT
CLRCEM CMDATA CMR	IP.BCFAP IP.MCPU IP.CMPID IP.INT5 IP.SLIB	IP.SIDLE IP.C176 IP.IUSID IP.SPT	IP.DCT IP.MCPU IP.SYSE	IP.ECFL IP.MECS IP.VER	IP.ECSB IP.MPPU IP.819	IP.FTHRL IP.SIDLE MODEL
	OS.NAME PR.SYS	OS.VER IP.NDEN	PR.BATCH IP.TDEN	PR.NUMB IP.TSG	PR.INT	PR.SCP
CMRDIR CMRTEXT	IP.SIDLE IP.ECSB PR.BATCH	IP.819 IP.MCPU PR.INT	IP.819 PR.SCP	MODEL IP.NDEN	OS.NAME IP.TDEN	OS.VER IP.TSG
CPECSM CPMTR	IP.MECS IP.C176 IP.SIDLE	IP.ECSB IP.SPT	IP.MCPU IP.TCPUB	IP.TCPUB:		
	IP.XJ					
CPSCH	IP.BRCL	IP.IRCL	IP.SIDLE	IP.SPT		
CPSM	IP.SPT	IP.TYPE				
CPSPM	IP.SPT	IP.819				
CPSS	IP.MCPU	IP.MECS				
CPUST	IP.MCPU	IP.XJ	PR.SYS			
CSWP	IP.DECR	IP.MXQT				
DIS	IP.MCPU PR.BATCH	IP.MECS PR.INT	IP.819 PR.SCP	MODEL	OS.NAME	OS.VER
DRVS	IP.CYSTP					
DSD	IP.C176 IP.MECS IP.YMD PR.INT	IP.ECSB IP.SCHDE IP.819 PR.SCP	IP.ILCMD IP.SFL MODEL IP.NDEN	IP.INT5 IP.SIDLE OS.NAME IP.TDEN	IP.LINK IP.STEX OS.VER IP.TSG	IP.MCPU IP.XJ PR.BATCH
ECSUR	IP.BCFAP					
EXBOOT	IP.ECSB					
FLUSHST	IP.CMBL					
GETRAND	IP.CMBL					
IPTEXT	IP.819 PR.SCP	MODEL	OS.NAME	OS.VER	PR.BATCH	PR.INT
IRCP	IP.ECSB OS.NAME	IP.SCHDE OS.VER	IP.SPT PR.BATCH	IP.YMD PR.INT	IP.819 PR.SCP	MODEL IP.EBUF
LABEL	IP.819 PR.SCP	MODEL IP.NDEN	OS.NAME IP.RCYC	OS.VER IP.TDEN	PR.BATCH IP.TSG	PR.INT IP.WEC
LINK	IP.MCPU	IP.TYPE				
MEM	IP.ECSB PR.BATCH	IP.MECS PR.INT	IP.819 PR.SCP	MODEL	OS.NAME	OS.VER
MTR	IP.C176 IP.SIDLE PR.BATCH	IP.ECSB IP.XJ PR.INT	IP.MCPU IP.819 PR.SCP	IP.MECS MODEL	IP.MPPU OS.NAME	IP.POSFL OS.VER
PACKAGE PPTXT	IP.MCPU IP.INT5					
QAC	IP.ECSB PR.INT	IP.819 PR.SCP	MODEL	OS.NAME	OS.VER	PR.BATCH
QAF	IP.ECSB PR.INT	IP.819 PR.SCP	MODEL	OS.NAME	OS.VER	PR.BATCH
RAGET REQ	IP.MCPU IP.ECSB PR.INT	IP.XJ IP.819 PR.SCP	MODEL IP.NDEN	OS.NAME IP.TDEN	OS.VER IP.TSG	PR.BATCH

REQEUBUF REQUEST	IP.CMBL IP.819 PR.SCP	MODEL IP.NDEN	OS.NAME IP.TDEN	OS.VER IP.TSG	PR.BATCH IP.WEC	PR.INT
RESCH	IP.MCPU IP.TYPE	IP.SIDLE IP.XJ	IP.TCPUB	IP.TCPUB:		
RPV	IP.CPLM OS.NAME	IP.IOLM OS.VER	IP.MCPU PR.BATCH	IP.MSLM PR.INT	IP.819 PR.SCP	MODEL
SCHRES SCPTXT SEGLINK SEGPARG SETST SPRBMGR SSC	IP.TYPE IP.INT5 IP.C176 IP.XJ IP.MCPU IP.MPPU IP.819 PR.SCP	IP.XJ				
SSCSEG SSFSEG SSH	IP.MCPU IP.MCPU IP.819 PR.SCP	MODEL	OS.NAME	OS.VER	PR.BATCH	PR.INT
STL	IP.ECSB PR.BATCH	IP.SRMS PR.INT	IP.819 PR.SCP	MODEL	OS.NAME	OS.VER
STORE STS	IP.CMBL IP.SFL PR.INT	IP.819 PR.SCP	MODEL	OS.NAME	OS.VER	PR.BATCH
SYSIDLE TDS	IP.MCPU IP.ECSB PR.BATCH	IP.SIDLE IP.SIDLE PR.INT	IP.819 PR.SCP	MODEL	OS.NAME	OS.VER
TIMSEG UPM USERMOD USERR WOR 1AB	IP.MCPU IP.MCPU IP.XJ IP.MCPU IP.DBAL PR.BATCH	IP.MXTIN	IP.ROFL			
1AJ	IP.ACNT MODEL	IP.ECSB OS.NAME	IP.IACES OS.VER	IP.MMS PR.BATCH	IP.SFL PR.INT	IP.819 PR.SCP
1BT 1CL	IP.NDEN IP.SCHDE PR.INT	IP.TDEN IP.819 PR.SCP	IP.TSG MODEL IP.NDEN	OS.NAME OS.VER IP.TSG	OS.VER OS.NAME IP.NTCN	PR.BATCH OS.VER IP.PCL
1CR	IP.C63 PR.BATCH	IP.CSET PR.INT	IP.819 PR.SCP	MODEL IP.NDEN	OS.NAME IP.NTCN	OS.VER IP.PCL
1CS	IP.C63 PR.BATCH	IP.CSET PR.INT	IP.819 PR.SCP	MODEL IP.NDEN	OS.NAME IP.NTCN	OS.VER IP.PCL
1CT	IP.PTCN IP.C63 PR.BATCH	IP.RPE1 IP.CSET PR.INT	IP.RPE2 IP.819 PR.SCP	IP.TDEN MODEL IP.NDEN	IP.TSG OS.NAME IP.NTCN	OS.VER OS.VER IP.PCL
1C9	IP.PTCN IP.C63 PR.BATCH	IP.RPE1 IP.CSET PR.INT	IP.RPE2 IP.819 PR.SCP	IP.TDEN MODEL IP.NDEN	IP.TSG OS.NAME IP.NTCN	OS.VER OS.VER IP.PCL
1EJ	IP.ECSB OS.NAME IP.NDEN	IP.LVF OS.VER IP.TDEN	IP.NDFS PR.BATCH IP.TSG	IP.SCHDE PR.INT IP.TSG	IP.819 PR.SCP	MODEL IP.NBCD
1IB	IP.BOVF IP.MECS MODEL	IP.DBAL IP.MSCT OS.NAME	IP.ECSB IP.SCHDE OS.VER	IP.IQPW IP.SMS PR.BATCH	IP.LVF IP.TOVF PR.INT	IP.MCPU IP.819 PR.SCP
1IT	IP.NDEN IP.CSET PR.INT IP.TSG	IP.TDEN IP.819 PR.SCP	IP.TSG MODEL IP.NDEN	OS.NAME OS.VER IP.NOISE	OS.VER IP.NOIS9	PR.BATCH IP.TDEN
1LC	IP.CSET PR.BATCH	IP.SCHDE PR.INT	IP.819 PR.SCP	MODEL	OS.NAME	OS.VER
1MF 1MH	IP.NDEN IP.819 PR.SCP	IP.TDEN MODEL	IP.TSG OS.NAME	OS.VER	PR.BATCH	PR.INT
1MT	IP.C63 PR.BATCH	IP.CSET PR.INT	IP.819 PR.SCP	MODEL IP.NDEN	OS.NAME IP.NOISE	OS.VER IP.NTCN
1NO	IP.PCL IP.C63 PR.BATCH IP.NTCN IP.TSG	IP.PTCN IP.CSET PR.INT IP.PCL	IP.RPE1 IP.819 PR.SCP IP.PTCN	IP.RPE2 MODEL IP.NDEN IP.RPE1	IP.TDEN OS.NAME IP.NOISE IP.RPE2	OS.VER OS.VER IP.NOIS9 IP.TDEN

1NR	IP.C63 PR.BATCH IP.NTCN IP.TSG	IP.CSET PR.INT IP.PCL	IP.819 PR.SCP IP.PTCN	MODEL IP.NDEN IP.RPE1	OS.NAME IP.NOISE IP.RPE2	OS.VER IP.NOIS9 IP.TDEN
1NW	IP.C63 PR.BATCH IP.NTCN IP.TSG	IP.CSET PR.INT IP.PCL	IP.819 PR.SCP IP.PTCN	MODEL IP.NDEN IP.RPE1	OS.NAME IP.NOISE IP.RPE2	OS.VER IP.NOIS9 IP.TDEN
1N2	IP.C63 PR.BATCH IP.NTCN IP.TSG	IP.CSET PR.INT IP.PCL	IP.819 PR.SCP IP.PTCN	MODEL IP.NDEN IP.RPE1	OS.NAME IP.NOISE IP.RPE2	OS.VER IP.NOIS9 IP.TDEN
1N3	IP.C63 PR.BATCH IP.NTCN IP.TSG	IP.CSET PR.INT IP.PCL	IP.819 PR.SCP IP.PTCN	MODEL IP.NDEN IP.RPE1	OS.NAME IP.NOISE IP.RPE2	OS.VER IP.NOIS9 IP.TDEN
1OP	IP.NDEN	IP.TDEN	IP.TSG			
1P1	IP.C63 PR.BATCH IP.NTCN IP.TSG	IP.CSET PR.INT IP.PCL	IP.819 PR.SCP IP.PTCN	MODEL IP.NDEN IP.RPE1	OS.NAME IP.NOISE IP.RPE2	OS.VER IP.NOIS9 IP.TDEN
1P2	IP.C63 PR.BATCH IP.PTCN	IP.CSET PR.INT IP.RPE1	IP.819 PR.SCP IP.RPE2	MODEL IP.NDEN IP.TDEN	OS.NAME IP.NTCN IP.TSG	OS.VER IP.PCL
1P3	IP.C63 PR.BATCH IP.PTCN	IP.CSET PR.INT IP.RPE1	IP.819 PR.SCP IP.RPE2	MODEL IP.NDEN IP.TDEN	OS.NAME IP.NTCN IP.TSG	OS.VER IP.PCL
1P4	IP.C63 PR.BATCH IP.PTCN	IP.CSET PR.INT IP.RPE1	IP.819 PR.SCP IP.RPE2	MODEL IP.NDEN IP.TDEN	OS.NAME IP.NTCN IP.TSG	OS.VER IP.PCL
1RN	IP.C63 IP.RBINT PR.BATCH IP.PTCN	IP.CSET IP.SRMS PR.INT IP.RPE1	IP.ECSB IP.819 PR.SCP IP.RPE2	MODEL IP.NDEN IP.TDEN	IP.LVF OS.NAME IP.NTCN IP.TSG	IP.00D OS.VER IP.PCL
1RP	IP.819 PR.SCP	IP.NDEN IP.NDEN	OS.NAME IP.TDEN	OS.VER IP.TSG	PR.BATCH	PR.INT
1RS	IP.C63 PR.BATCH IP.PCL	IP.CSET PR.INT IP.PTCN	IP.819 PR.SCP IP.RPE1	MODEL IP.NDEN IP.RPE2	OS.NAME IP.NOISE IP.TDEN	OS.VER IP.NTCN IP.TSG
1RT	IP.C63 PR.BATCH IP.PTCN	IP.CSET PR.INT IP.RPE1	IP.819 PR.SCP IP.RPE2	MODEL IP.NDEN IP.TDEN	OS.NAME IP.NTCN IP.TSG	OS.VER IP.PCL
1RV	IP.C63 PR.BATCH IP.PTCN	IP.CSET PR.INT IP.RPE1	IP.819 PR.SCP IP.RPE2	MODEL IP.NDEN IP.TDEN	OS.NAME IP.NTCN IP.TSG	OS.VER IP.PCL
1R2	IP.C63 PR.BATCH IP.NTCN IP.TSG	IP.CSET PR.INT IP.PCL	IP.819 PR.SCP IP.PTCN	MODEL IP.NDEN IP.RPE1	OS.NAME IP.NOISE IP.RPE2	OS.VER IP.NOIS9 IP.TDEN
1R3	IP.C63 PR.BATCH IP.NTCN IP.TSG	IP.CSET PR.INT IP.PCL	IP.819 PR.SCP IP.PTCN	MODEL IP.NDEN IP.RPE1	OS.NAME IP.NOISE IP.RPE2	OS.VER IP.NOIS9 IP.TDEN
1R9	IP.C63 PR.BATCH IP.PTCN	IP.CSET PR.INT IP.RPE1	IP.819 PR.SCP IP.RPE2	MODEL IP.NDEN IP.TDEN	OS.NAME IP.NTCN IP.TSG	OS.VER IP.PCL
1SC	IP.C176 PR.BATCH	IP.SIDLE PR.INT	IP.819 PR.SCP	MODEL	OS.NAME	OS.VER
1SI	IP.ECSB MODEL	IP.MCPU OS.NAME	IP.MECS OS.VER	IP.SFL PR.BATCH	IP.SMS PR.INT	IP.819 PR.SCP
1S0	IP.ECSB IP.819 PR.SCP	IP.MCPU MODEL	IP.MECS OS.NAME	IP.POSFL OS.VER	IP.ROFL PR.BATCH	IP.SPT PR.INT
1SP	IP.ECSB PR.BATCH	IP.MCPU PR.INT	IP.819 PR.SCP	MODEL	OS.NAME	OS.VER
1S5	IP.819 PR.SCP	MODEL	OS.NAME	OS.VER	PR.BATCH	PR.INT
1TF	IP.C63 PR.BATCH IP.NTCN IP.TSG	IP.CSET PR.INT IP.PCL	IP.819 PR.SCP IP.PTCN	MODEL IP.NDEN IP.RPE1	OS.NAME IP.NOISE IP.RPE2	OS.VER IP.NOIS9 IP.TDEN
1TJ	IP.LVF IP.SFL OS.NAME IP.TDEN	IP.MCPU IP.SIOL OS.VER IP.TSG	IP.MECS IP.SPR PR.BATCH	IP.MPR IP.STL PR.INT	IP.MTL IP.819 PR.SCP	IP.SCHDE MODEL IP.NDEN

1TO	IP.819	MODEL	OS.NAME	OS.VER	PR.BATCH	PR.INT
	PR.SCP	IP.NDEN	IP.TDEN	IP.TSG		
1TR	IP.819	MODEL	OS.NAME	OS.VER	PR.BATCH	PR.INT
	PR.SCP	IP.NDEN	IP.TDEN	IP.TSG		
1TS	IP.SCHDE	IP.819	MODEL	OS.NAME	OS.VER	PR.BATCH
	PR.INT	PR.SCP	IP.NBCD	IP.NDEN	IP.TDEN	IP.TSG
1WI	IP.C63	IP.CSET	IP.819	MODEL	OS.NAME	OS.VER
	PR.BATCH	PR.INT	PR.SCP	IP.NDEN	IP.NTCN	IP.PCL
	IP.PTCN	IP.RPE1	IP.RPE2	IP.TDEN	IP.TSG	
1WS	IP.C63	IP.CSET	IP.819	MODEL	OS.NAME	OS.VER
	PR.BATCH	PR.INT	PR.SCP	IP.NDEN	IP.NOISE	IP.NTCN
	IP.PCL	IP.PTCN	IP.RPE1	IP.RPE2	IP.TDEN	IP.TSG
1W9	IP.C63	IP.CSET	IP.819	MODEL	OS.NAME	OS.VER
	PR.BATCH	PR.INT	PR.SCP	IP.NDEN	IP.NTCN	IP.PCL
	IP.PTCN	IP.RPE1	IP.RPE2	IP.TDEN	IP.TSG	
2IA	IP.CSET	IP.819	MODEL	OS.NAME	OS.VER	PR.BATCH
	PR.INT	PR.SCP	IP.NDEN	IP.TDEN	IP.TSG	
2IB	IP.819	MODEL	OS.NAME	OS.VER	PR.BATCH	PR.INT
	PR.SCP	IP.NDEN	IP.TDEN	IP.TSG		
2IC	IP.CSET	IP.819	MODEL	OS.NAME	OS.VER	PR.BATCH
	PR.INT	PR.SCP	IP.NDEN	IP.TDEN	IP.TSG	
2ID	IP.CSET	IP.819	MODEL	OS.NAME	OS.VER	PR.BATCH
	PR.INT	PR.SCP	IP.NDEN	IP.TDEN	IP.TSG	
2IL	IP.819	MODEL	OS.NAME	OS.VER	PR.BATCH	PR.INT
	PR.SCP	IP.NDEN	IP.TDEN	IP.TSG		
2IO	IP.819	MODEL	OS.NAME	OS.VER	PR.BATCH	PR.INT
	PR.SCP	IP.NDEN	IP.RCYC	IP.TDEN	IP.TSG	
2IP	IP.CSET	IP.819	MODEL	OS.NAME	OS.VER	PR.BATCH
	PR.INT	PR.SCP	IP.NDEN	IP.TDEN	IP.TSG	
2IR	IP.CSET					
2IT	IP.CSET	IP.819	MODEL	OS.NAME	OS.VER	PR.BATCH
	PR.INT	PR.SCP	IP.NDEN	IP.TDEN	IP.TSG	
2IW	IP.CSET	IP.819	MODEL	OS.NAME	OS.VER	PR.BATCH
	PR.INT	PR.SCP	IP.NDEN	IP.TDEN	IP.TSG	
2IX	IP.CSET	IP.819	MODEL	OS.NAME	OS.VER	PR.BATCH
	PR.INT	PR.SCP	IP.NDEN	IP.TDEN	IP.TSG	
2TB	IP.C63	IP.CSET	IP.819	MODEL	OS.NAME	OS.VER
	PR.BATCH	PR.INT	PR.SCP	IP.NDEN	IP.NOISE	IP.NOIS9
	IP.NTCN	IP.PCL	IP.PTCN	IP.RPE1	IP.RPE2	IP.TDEN
	IP.TSG					
2TC	IP.NDEN	IP.TDEN	IP.TSG			
2VJ	IP.LVF	IP.MCPU	IP.MECS	IP.MPR	IP.MTL	IP.SCHDE
	IP.SFL	IP.SIOL	IP.SPR	IP.STL	IP.819	MODEL
	OS.NAME	OS.VER	PR.BATCH	PR.INT	PR.SCP	IP.NDEN
	IP.TDEN	IP.TSG				
300	IP.ECSB	IP.819	MODEL	OS.NAME	OS.VER	PR.BATCH
	PR.INT	PR.SCP	IP.NDEN	IP.TDEN	IP.TSG	
3IC	IP.SCHDE	IP.819	MODEL	OS.NAME	OS.VER	PR.BATCH
	PR.INT	PR.SCP	IP.NDEN	IP.TDEN	IP.TSG	
3IE	IP.SCHDE	IP.819	MODEL	OS.NAME	OS.VER	PR.BATCH
	PR.INT	PR.SCP	IP.NDEN	IP.TDEN	IP.TSG	
3IF	IP.819	MODEL	OS.NAME	OS.VER	PR.BATCH	PR.INT
	PR.SCP	IP.NDEN	IP.TDEN	IP.TSG		
3II	IP.CSET	IP.NOISE	IP.NOIS9			
3IJ	IP.819	MODEL	OS.NAME	OS.VER	PR.BATCH	PR.INT
	PR.SCP	IP.NDEN	IP.TDEN	IP.TSG		
3IL	IP.819	MODEL	OS.NAME	OS.VER	PR.BATCH	PR.INT
	PR.SCP	IP.NDEN	IP.RCYC	IP.TDEN	IP.TSG	
3IM	IP.819	MODEL	OS.NAME	OS.VER	PR.BATCH	PR.INT
	PR.SCP	IP.NDEN	IP.TDEN	IP.TSG		
3IN	IP.819	MODEL	OS.NAME	OS.VER	PR.BATCH	PR.INT
	PR.SCP	IP.NDEN	IP.TDEN	IP.TSG		
3IO	IP.819	MODEL	OS.NAME	OS.VER	PR.BATCH	PR.INT
	PR.SCP	IP.NDEN	IP.TDEN	IP.TSG		
3IR	IP.SCHDE	IP.819	MODEL	OS.NAME	OS.VER	PR.BATCH
	PR.INT	PR.SCP	IP.NDEN	IP.TDEN	IP.TSG	
3IV	IP.819	MODEL	OS.NAME	OS.VER	PR.BATCH	PR.INT
	PR.SCP	IP.NDEN	IP.TDEN	IP.TSG		
3IW	IP.SCHDE	IP.819	MODEL	OS.NAME	OS.VER	PR.BATCH
	PR.INT	PR.SCP	IP.NDEN	IP.TDEN	IP.TSG	
3MN	IP.SCHDE	IP.NBCD	IP.NBRK	IP.NDEN	IP.TDEN	
3RQ	IP.ECSR	IP.SCHDE	IP.819	MODEL	OS.NAME	OS.VER
	PR.BATCH	PR.INT	PR.SCP	IP.NDEN	IP.TDEN	
4DO	IP.NDEN	IP.TDEN	IP.TSG			
4EJ	IP.LVF	IP.MCPU	IP.MECS	IP.MPR	IP.MTL	IP.SCHDE



	IP.SFL	IP.SIOL	IP.SPR	IP.STL	IP.NDEN		
4ES	IP.NDEN	IP.TDEN	IP.TSG				
4LB	IP.C63	IP.CSET	IP.819	MODEL	OS.NAME	OS.VER	
	PR.BATCH	PR.INT	PR.SCP	IP.NDEN	IP.NTCN	IP.PCL	
	IP.PTCN	IP.RCYC	IP.RPE1	IP.RPE2	IP.TDEN	IP.TSG	
4LC	IP.C63	IP.CSET	IP.819	MODEL	OS.NAME	OS.VER	
	PR.BATCH	PR.INT	PR.SCP	IP.NDEN	IP.NTCN	IP.PCL	
	IP.PTCN	IP.RCYC	IP.RPE1	IP.RPE2	IP.TDEN	IP.TSG	
6LC	IP.NBCD	IP.PTCN					
6L3	IP.NBCD						
6RD	IP.LVF	IP.OPRI	IP.SPR	IP.819	MODEL	OS.NAME	
	OS.VER	PR.BATCH	PR.INT	PR.SCP			
6WM	IP.819	MODEL	OS.NAME	OS.VER	PR.BATCH	PR.INT	
	PR.SCP	IP.NDEN	IP.TDEN	IP.TSG			
7AJ	IP.CPLM	IP.IOLM	IP.MCPU	IP.MSLM			
7EC	IP.ECSB	IP.819					
7T1	IP.CSET						
7T2	IP.CSET						
8T3	IP.C63	IP.CSET	IP.819	MODEL	OS.NAME	OS.VER	
	PR.BATCH	PR.INT	PR.SCP	IP.NBCD	IP.NDEN	IP.NTCN	
	IP.PCL	IP.PTCN	IP.RPE1	IP.RPE2	IP.TDEN	IP.TSG	

\*\*\*\*PL1B\*\*\*\*

\*\*\*\*\*

IPARAMS

\*\*\*\*\*

IP.ARCH	PFA	PFP					
IP.CP	1IS						
IP.CR	1IR	2IS	3IS				
IP.CSET	1IR	1IS	3IS	4IS	CVL		
IP.ECSB	LDD	LDQ	LDV	LDW			
IP.IACES	GCC						
IP.INT5	RWE						
IP.LVF	IPF						
IP.MSCT	MSG						
IP.PFRP	IPC	PFC	1QF	SPF	1FC	PFA	
	GPF	LPF	PFP	PFE	PFR	PFS	
	OUX						
IP.RM	MNT	RELABEL	LABELMS				
IP.SRMS	MNT						
IP.TYPE	CVL						
IP.US	IPC	PFC	1QF	SPF	1FC	PFA	
	GPF	LPF	PFP	PFE	PFR	PFS	
	OUX						
IP.UP	PFA						
IP.YMD	1GM	DAYSWAP					
OS.ID	LOADO	LOADO3					
PR.BATCH	HDS						

IPARAMS SORTED BY ROUTINE

CVL	IP.CSET	IP.TYPE		
DAYSWAP	IP.YMD			
GCC	IP.IACES			
GPF	IP.PFRP	IP.US		
HDS	PR.BATCH			
LABELMS	IP.RM			
LDD	IP.ECSB			
LDQ	IP.ECSB			
LDV	IP.ECSB			
LDW	IP.ECSB			
LOADO	OS.ID			
LOADO3	OS.ID			
LPF	IP.PFRP	IP.US		
MNT	IP.RM	IP.SRMS		
MSG	IP.MSCT			
OUX	IP.PFRP	IP.US		
PFA	IP.ARCH	IP.PFRP	IP.US	IP.UP
PFC	IP.PFRP	IP.US		
PFE	IP.PFRP	IP.US		
PFP	IP.ARCH	IP.PFRP	IP.US	
PFR	IP.PFRP	IP.US		
PFS	IP.PFRP	IP.US		
RELABEL	IP.RM			
RWE	IP.INT5			

SPF	IP.PFRP	IP.US
1FC	IP.PFRP	IP.US
1GM	IP.YMD	
1IR	IP.CR	IP.CSET
1IS	IP.CP	IP.CSET
1PC	IP.PFRP	IP.US
1PF	IP.LVF	
1QF	IP.PFRP	IP.US
2IS	IP.CR	
3IS	IP.CR	IP.CSET
4IS	IP.CSET	

\*\*\*\*UPDATE\*\*\*\*

\*\*\*\*\*

IPARAMS

\*\*\*\*\*

IP.C63	UPDATE		
IP.PD	ITEMIZE		
MODEL	UPDATE		
OS.NAME	COPYL	ITEMIZE	UPDATE
OS.VER	COPYL	ITEMIZE	UPDATE

IPARAMS SORTED BY ROUTINE

COPYL	OS.NAME	OS.VER		
ITEMIZE	IP.PD	OS.NAME	OS.VER	
UPDATE	IP.C63	MODEL	OS.NAME	OS.VER

\*\*\*\*CYBER LOADER\*\*\*\*

\*\*\*\*\*

IPARAMS

\*\*\*\*\*

IP.IACES	LOADER	LOADZ	LOADU			
IP.MECS	LOADER	LOADG	LOADS	LOADZ	LOADM	LOADU
	LOADUM	UCLOAD				
IP.PD	LOADER	LOADC	LOADS	LOADZ	LOADM	LOADU
	LOADUC	LOADUM				
IP.PS	LOADER	LOADC	LOADS	LOADZ	LOADM	LOADU
	LOADUC	LOADUM				
OS.ID	LOADER	LOADZ	LOADU			

IPARAMS SORTED BY ROUTINE

LOADC	IP.PD	IP.PS			
LOADER	IP.IACES	IP.MECS	IP.PD	IP.PS	OS.ID
LOADG	IP.MECS				
LOADM	IP.MECS	IP.PD	IP.PS		
LOADS	IP.MECS	IP.PD	IP.PS		
LOADU	IP.IACES	IP.MECS	IP.PD	IP.PS	OS.ID
LOADUC	IP.PD	IP.PS			
LOADUM	IP.MECS	IP.PD	IP.PS		
LOADZ	IP.IACES	IP.MECS	IP.PD	IP.PS	OS.ID
UCLOAD	IP.MECS				

\*\*\*\*819 SUPPORT\*\*\*\*

\*\*\*\*\*

IPARAMS

\*\*\*\*\*

IP.BCFAP	HLOG	HACT
----------	------	------

IPARAMS SORTED BY ROUTINE

HACT	IP.BCFAP
HLOG	IP.BCFAP

\*\*\*\*COMPASS\*\*\*\*

\*\*\*\*\*

IPARAMS

\*\*\*\*\*

IP.PD	COMPASS	"CP.NAME"
IP.PS	COMPASS	"CP.NAME"
MODEL	COMPASS	"CP.NAME"
OS.NAME	COMPASS	
OS.ID	"CP.NAME"	
OS.VER	COMPASS	

\*\*\*\*\*  
 CIOCOM  
 \*\*\*\*\*  
 ECSCOM  
 \*\*\*\*\*

	IPARAMS	SORTED BY		ROUTINE		
"CP.NAME"	IP.PD	IP.PS	MODEL	OS.ID		
COMPASS	IP.PD	IP.PS	MODEL	OS.NAME	OS.VER	

\*\*\*\*BASIC ACCESS METHODS\*\*\*\*

\*\*\*\*\*  
 IPARAMS  
 \*\*\*\*\*  
 IP.CMU                   TXTCRM  
 OS.NAME                 IOTEXT        SYSTEXT     TXTCRM

	IPARAMS	SORTED BY		ROUTINE
IOTEXT	OS.NAME			
SYSTEXT	OS.NAME			
TXTCRM	IP.CMU	OS.NAME		

\*\*\*\*ADVANCED ACCESS METHODS INITIAL\*\*\*\*

\*\*\*\*\*  
 IPARAMS  
 \*\*\*\*\*  
 IP.CSET                 SISOPEN       RMSMIP  
 OS.NAME                 PPCALL        RMSMCAL

	IPARAMS	SORTED BY		ROUTINE
PPCALL	OS.NAME			
RMSMCAL	OS.NAME			
RMSMIP	IP.CSET			
SISOPEN	IP.CSET			

\*\*\*\*ADVANCED ACCESS METHODS EXTENDED\*\*\*\*

\*\*\*\*\*  
 IPARAMS  
 \*\*\*\*\*  
 IP.CSET                 DICO\$AA  
 OS.NAME                 CRA1\$AA

	IPARAMS	SORTED BY		ROUTINE
CRA1\$AA	OS.NAME			
DICO\$AA	IP.CSET			

\*\*\*\*BIT 8\*\*\*\*

\*\*\*\*\*  
 IPARAMS  
 \*\*\*\*\*  
 IP.C63                   COPY8P  
 IP.CSET                 T8.HXTB       BDPTAB       T8.6TAB     COPY8P

	IPARAMS	SORTED BY		ROUTINE
BDPTAB	IP.CSET			
COPY8P	IP.C63	IP.CSET		
T8.HXTB	IP.CSET			
T8.6TAB	IP.CSET			

\*\*\*\*CE DIAGS\*\*\*\*

\*\*\*\*\*  
 IPARAMS  
 \*\*\*\*\*  
 IP.YMD                   NORMSS

\*\*\*\*\*  
CIOCOM  
\*\*\*\*\*

IPARAMS SORTED BY ROUTINE

NORMSS IP.YMD

\*\*\*\*SYMPL\*\*\*\*

\*\*\*\*\*  
IPARAMS  
\*\*\*\*\*  
OS.ID  
\*\*\*\*\*  
CIOCOM  
\*\*\*\*\*  
ECSCOM  
\*\*\*\*\*

INIT14 INIT40

IPARAMS SORTED BY ROUTINE

INIT14 OS.ID  
INIT40 OS.ID

\*\*\*\*FTN COMPILER\*\*\*\*

\*\*\*\*\*  
IPARAMS  
\*\*\*\*\*  
IP.PD  
IP.PS  
MODEL  
OS.NAME  
OS.VER

TSTEXT FTNMAC FTNTEXT  
TSTEXT FTNMAC FTNTEXT  
TSTEXT FTNMAC FTNTEXT  
TSTEXT FTNMAC FTNTEXT  
TSTEXT FTNMAC FTNTEXT

IPARAMS SORTED BY ROUTINE

FTNMAC	IP.PD	IP.PS	MODEL	OS.NAME	OS.VER
FTNTEXT	IP.PD	IP.PS	MODEL	OS.NAME	OS.VER
TSTEXT	IP.PD	IP.PS	MODEL	OS.NAME	OS.VER

\*\*\*\*FTN LIBRARY\*\*\*\*

\*\*\*\*\*  
IPARAMS  
\*\*\*\*\*  
MODEL

Q2NTRY=

IPARAMS SORTED BY ROUTINE

Q2NTRY= MODEL

\*\*\*\*COBOL 4\*\*\*\*

\*\*\*\*\*  
IPARAMS  
\*\*\*\*\*  
IP.CMU

D=SUBMV	D=OPWA	C=COBIO	D=SBLIT	D=KYDEF	C=CS1P0
D=CLS	D=DEL	D=OPN	D=RD2	D=REW	D=WR2
D=RST	D=RELR	D=SQIO	D=DAIO	D=START	D=ISIO
D=AKIO	D=UPCNT	D=ADVAN	D=SUBSC	D=SBSC2	D=SBSC6
D=SBSC7	D=SOL	D=SORT	D=DSPLY	D=XCEPT	D=DADD
C=NSTAN	D=TRUBL	D=CVBD	D=NISA	D=BN	D=N6SA
D=N6SS	B=1DSA	D=N1DS	R=6DSA	D=N6DS	D=1DDA
D=N1DD	D=6DDA	D=N6DD	D=N1SB	B=1SBA	D=EXAMO
D=LITCU	D=LITCC	D=BCDCC	D=COLSQ	D=BCDCU	D=BCDCM
D=SCM	C=NVI	CONTROL	SAVEREG	DIAG	SNAP
BLOCKIO	SCAN2	ITEMCOP	SAD	LEXDATA	PASS1B
O1B	CONCRDI	LEXXY	PASS1B1	O1B1	PASS1C
PASS1D	LEXPROC	PASS1E	DIG	O1E	REF

	PASS1F	ART	MNEMON	PASS1G	TROUT1G	DISCIO
	PASS1H	GOTOGEN	ALTRGEN	GENPRFM	GENPLIM	PRFOPS
	TCLIMB	GENLOD	GENSTO	PASS2	CHECK	GENARTH
	GENMOVE	GENIF	LIT02	GENDISP	SUBSCR	GREPORT
IP.CSET	DIAGOUT	DAGMSG	CREF			
IP.PD	D=SUBMV	D=OPWA	D=COLSQ	C=NV1	ART	C=CS1P0
	D=SUBMV	D=DEL	C=COBIO	D=SBLIT	D=KYDEF	D=WR2
	D=CLS	D=REL	D=OPN	D=RD2	D=REW	D=ISIO
	D=RST	D=UPCNT	D=SQIO	D=DAIO	D=START	D=SBSC6
	D=AKIO	D=SOL	D=ADVAN	D=SUBSC	D=SBSC2	D=DADD
	D=SBSC7	D=TRUBL	D=CVBD	D=DSPLY	D=XCEPT	D=N6SA
	C=NSTAN	B=1DSA	D=N1DS	D=N1SA	D=BN	D=1DDA
	D=N6SS	D=6DDA	D=N6DD	R=6DSA	D=N6DS	D=EXAMO
	D=N1DD	D=LITCC	D=BCDCC	D=N1SB	B=1SBA	D=BCDCM
	D=LITCU	C=NV1	CONTROL	D=COLSQ	D=BCDCU	DIAG
	D=SCM	SCAN2	ITEMCOP	SAVEREG	DIAG	SNAP
	BLOCKIO	CONCRDI	LEXXY	SAD	LEXDATA	PASS1B
	01B	LEXPROC	PASS1E	PASS1B1	01B1	PASS1C
	PASS1D	ART	MNEMON	DIG	01E	REF
	PASS1F	GOTOGEN	ALTRGEN	PASS1G	TROUT1G	DISCIO
	PASS1H	GENLOD	GENSTO	GENPRFM	GENPLIM	PRFOPS
	TCLIMB	GENIF	LIT02	PASS2	CHECK	GENARTH
	GENMOVE	DAGMSG	CREF	GENDISP	SUBSCR	GREPORT
	DIAGOUT	D=OPWA	C=COBIO	D=SBLIT	D=KYDEF	C=CS1P0
IP.PS	D=SUBMV	D=DEL	D=OPN	D=RD2	D=REW	D=WR2
	D=CLS	D=REL	D=SQIO	D=DAIO	D=START	D=ISIO
	D=RST	D=UPCNT	D=ADVAN	D=SUBSC	D=SBSC2	D=SBSC6
	D=AKIO	D=SOL	D=CVBD	D=DSPLY	D=XCEPT	D=DADD
	D=SBSC7	D=TRUBL	D=N1DS	D=N1SA	D=BN	D=N6SA
	C=NSTAN	B=1DSA	D=N6DD	B=6DSA	D=N6DS	D=1DDA
	D=N6SS	D=6DDA	D=BCDCC	D=N1SB	B=1SBA	D=EXAMO
	D=N1DD	D=LITCC	CONTROL	D=COLSQ	D=BCDCU	D=BCDCM
	D=LITCU	C=NV1	ITEMCOP	SAVEREG	DIAG	SNAP
	D=SCM	SCAN2	LEXXY	SAD	LEXDATA	PASS1B
	BLOCKIO	CONCRDI	PASS1E	PASS1B1	01B1	PASS1C
	01B	LEXPROC	MNEMON	DIG	01E	REF
	PASS1D	ART	ALTRGEN	PASS1G	TROUT1G	DISCIO
	PASS1F	GOTOGEN	GENSTO	GENPRFM	GENPLIM	PRFOPS
	PASS1H	GENLOD	LIT02	PASS2	CHECK	GENARTH
	TCLIMB	GENIF	CREF	GENDISP	SUBSCR	GREPORT
	GENMOVE	DAGMSG				
	DIAGOUT	D=OPWA	C=COBIO	D=SBLIT	D=KYDEF	C=CS1P0
IP.YMD	D=SUBMV	D=DEL	D=OPN	D=RD2	D=REW	D=WR2
MODEL	D=CLS	D=REL	D=SQIO	D=DAIO	D=START	D=ISIO
	D=RST	D=UPCNT	D=ADVAN	D=SUBSC	D=SBSC2	D=SBSC6
	D=AKIO	D=SOL	D=CVBD	D=DSPLY	D=XCEPT	D=DADD
	D=SBSC7	D=TRUBL	D=N1DS	D=N1SA	D=BN	D=N6SA
	C=NSTAN	B=1DSA	D=N6DD	R=6DSA	D=N6DS	D=1DDA
	D=N6SS	D=6DDA	D=BCDCC	D=N1SB	B=1SBA	D=EXAMO
	D=N1DD	D=LITCC	CONTROL	D=COLSQ	D=BCDCU	D=BCDCM
	D=LITCU	C=NV1	ITEMCOP	SAVEREG	DIAG	SNAP
	D=SCM	SCAN2	LEXXY	SAD	LEXDATA	PASS1B
	BLOCKIO	CONCRDI	PASS1E	PASS1B1	01B1	PASS1C
	01B	LEXPROC	MNEMON	DIG	01E	REF
	PASS1D	ART	ALTRGEN	PASS1G	TROUT1G	DISCIO
	PASS1F	GOTOGEN	GENSTO	GENPRFM	GENPLIM	PRFOPS
	PASS1H	GENLOD	LIT02	PASS2	CHECK	GENARTH
	TCLIMB	GENIF	CREF	GENDISP	SUBSCR	GREPORT
	GENMOVE	DAGMSG				
	DIAGOUT	D=OPWA	C=COBIO	D=SBLIT	D=KYDEF	C=CS1P0
OS.NAME	D=SUBMV	D=DEL	D=OPN	D=RD2	D=REW	D=WR2
	D=CLS	D=REL	D=SQIO	D=DAIO	D=START	D=ISIO
	D=RST	D=UPCNT	D=ADVAN	D=SUBSC	D=SBSC2	D=SBSC6
	D=AKIO	D=SOL	D=CVBD	D=DSPLY	D=XCEPT	D=DADD
	D=SBSC7	D=TRUBL	D=N1DS	D=N1SA	D=BN	D=N6SA
	C=NSTAN	B=1DSA	D=N6DD	R=6DSA	D=N6DS	D=1DDA
	D=N6SS	D=6DDA	D=BCDCC	D=N1SB	B=1SBA	D=EXAMO
	D=N1DD	D=LITCC	CONTROL	D=COLSQ	D=BCDCU	D=BCDCM
	D=LITCU	C=NV1	ITEMCOP	SAVEREG	DIAG	SNAP
	D=SCM	SCAN2	LEXXY	SAD	LEXDATA	PASS1B
	BLOCKIO	CONCRDI	PASS1E	PASS1B1	01B1	PASS1C
	01B	LEXPROC	MNEMON	DIG	01E	REF
	PASS1D	ART	ALTRGEN	PASS1G	TROUT1G	DISCIO

PASSIF	ART	MNEMON	PASS1G	TROUT1G	DISCIO
PASS1H	GOTOGEN	ALTRGEN	GENPRFM	GENPLIM	PRFOPS
TCLIMB	GENLOD	GENSTO	PASS2	CHECK	GENARTH
GENMOVE	GENIF	LIT02	GENDISP	SUBSCR	GREPORT
DIAGOUT	DAGMSG	CREF			
ART					

OS.ID

IPARAMS SORTED BY ROUTINE

ALTRGEN	IP.CMU	IP.PD	IP.PS	MODEL	OS.NAME	
ART	IP.CMU	IP.CSET	IP.PD	IP.PS	MODEL	OS.NAME
OS.ID						
B=1DSA	IP.CMU	IP.PD	IP.PS	MODEL	OS.NAME	
B=1SBA	IP.CMU	IP.PD	IP.PS	MODEL	OS.NAME	
B=6DSA	IP.CMU	IP.PD	IP.PS	MODEL	OS.NAME	
BLOCKIO	IP.CMU	IP.PD	IP.PS	MODEL	OS.NAME	
C=COBIO	IP.CMU	IP.PD	IP.PS	MODEL	OS.NAME	
C=CS1P0	IP.CMU	IP.PD	IP.PS	MODEL	OS.NAME	
C=NSTAN	IP.CMU	IP.PD	IP.PS	MODEL	OS.NAME	
C=NV1	IP.CMU	IP.CSET	IP.PD	IP.PS	MODEL	OS.NAME
CHECK	IP.CMU	IP.PD	IP.PS	MODEL	OS.NAME	
CONCRDI	IP.CMU	IP.PD	IP.PS	MODEL	OS.NAME	
CONTROL	IP.CMU	IP.PD	IP.PS	MODEL	OS.NAME	
CREF	IP.CMU	IP.PD	IP.PS	MODEL	OS.NAME	
D=ADVAN	IP.CMU	IP.PD	IP.PS	MODEL	OS.NAME	
D=AKIO	IP.CMU	IP.PD	IP.PS	MODEL	OS.NAME	
D=BCDCC	IP.CMU	IP.PD	IP.PS	MODEL	OS.NAME	
D=BCDCM	IP.CMU	IP.PD	IP.PS	MODEL	OS.NAME	
D=BCDCU	IP.CMU	IP.PD	IP.PS	MODEL	OS.NAME	
D=BN	IP.CMU	IP.PD	IP.PS	MODEL	OS.NAME	
D=CLS	IP.CMU	IP.PD	IP.PS	MODEL	OS.NAME	
D=COLSQ	IP.CMU	IP.CSET	IP.PD	IP.PS	MODEL	OS.NAME
D=CVBD	IP.CMU	IP.PD	IP.PS	MODEL	OS.NAME	
D=DADD	IP.CMU	IP.PD	IP.PS	MODEL	OS.NAME	
D=DAIO	IP.CMU	IP.PD	IP.PS	MODEL	OS.NAME	
D=DEL	IP.CMU	IP.PD	IP.PS	MODEL	OS.NAME	
D=DSPLY	IP.CMU	IP.PD	IP.PS	MODEL	OS.NAME	
D=EXAMO	IP.CMU	IP.PD	IP.PS	MODEL	OS.NAME	
D=ISIO	IP.CMU	IP.PD	IP.PS	MODEL	OS.NAME	
D=KYDEF	IP.CMU	IP.PD	IP.PS	MODEL	OS.NAME	
D=LITCC	IP.CMU	IP.PD	IP.PS	MODEL	OS.NAME	
D=LITCU	IP.CMU	IP.PD	IP.PS	MODEL	OS.NAME	
D=N1DD	IP.CMU	IP.PD	IP.PS	MODEL	OS.NAME	
D=N1DS	IP.CMU	IP.PD	IP.PS	MODEL	OS.NAME	
D=N1SA	IP.CMU	IP.PD	IP.PS	MODEL	OS.NAME	
D=N1SB	IP.CMU	IP.PD	IP.PS	MODEL	OS.NAME	
D=N6DD	IP.CMU	IP.PD	IP.PS	MODEL	OS.NAME	
D=N6DS	IP.CMU	IP.PD	IP.PS	MODEL	OS.NAME	
D=N6SA	IP.CMU	IP.PD	IP.PS	MODEL	OS.NAME	
D=N6SS	IP.CMU	IP.PD	IP.PS	MODEL	OS.NAME	
D=OPN	IP.CMU	IP.PD	IP.PS	MODEL	OS.NAME	
D=OPWA	IP.CMU	IP.PD	IP.PS	MODEL	OS.NAME	
D=RD2	IP.CMU	IP.PD	IP.PS	MODEL	OS.NAME	
D=RELR	IP.CMU	IP.PD	IP.PS	MODEL	OS.NAME	
D=REW	IP.CMU	IP.PD	IP.PS	MODEL	OS.NAME	
D=RST	IP.CMU	IP.PD	IP.PS	MODEL	OS.NAME	
D=SBLIT	IP.CMU	IP.PD	IP.PS	MODEL	OS.NAME	
D=SBSC2	IP.CMU	IP.PD	IP.PS	MODEL	OS.NAME	
D=SBSC6	IP.CMU	IP.PD	IP.PS	MODEL	OS.NAME	
D=SBSC7	IP.CMU	IP.PD	IP.PS	MODEL	OS.NAME	
D=SCM	IP.CMU	IP.PD	IP.PS	MODEL	OS.NAME	
D=SOL	IP.CMU	IP.PD	IP.PS	MODEL	OS.NAME	
D=SORT	IP.CMU	IP.CSET	IP.PD	IP.PS	MODEL	OS.NAME
D=SQIO	IP.CMU	IP.PD	IP.PS	MODEL	OS.NAME	
D=START	IP.CMU	IP.PD	IP.PS	MODEL	OS.NAME	
D=SUBMV	IP.CMU	IP.CSET	IP.PD	IP.PS	MODEL	OS.NAME
D=SUBSC	IP.CMU	IP.PD	IP.PS	MODEL	OS.NAME	
D=TRUBL	IP.CMU	IP.PD	IP.PS	MODEL	OS.NAME	
D=UPCNT	IP.CMU	IP.PD	IP.PS	MODEL	OS.NAME	
D=WR2	IP.CMU	IP.PD	IP.PS	MODEL	OS.NAME	
D=XCEPT	IP.CMU	IP.PD	IP.PS	MODEL	OS.NAME	
D=XCPT	IP.YMD					
D=1DDA	IP.CMU	IP.PD	IP.PS	MODEL	OS.NAME	
D=6DDA	IP.CMU	IP.PD	IP.PS	MODEL	OS.NAME	

DAGMSG	IP.CMU	IP.PD	IP.PS	MODEL	OS.NAME
DIAG	IP.CMU	IP.PD	IP.PS	MODEL	OS.NAME
DIAGOUT	IP.CMU	IP.PD	IP.PS	MODEL	OS.NAME
DIG	IP.CMU	IP.PD	IP.PS	MODEL	OS.NAME
DISCIO	IP.CMU	IP.PD	IP.PS	MODEL	OS.NAME
GENARTH	IP.CMU	IP.PD	IP.PS	MODEL	OS.NAME
GENDISP	IP.CMU	IP.PD	IP.PS	MODEL	OS.NAME
GENIF	IP.CMU	IP.PD	IP.PS	MODEL	OS.NAME
GENLOD	IP.CMU	IP.PD	IP.PS	MODEL	OS.NAME
GENMOVE	IP.CMU	IP.PD	IP.PS	MODEL	OS.NAME
GENPLIM	IP.CMU	IP.PD	IP.PS	MODEL	OS.NAME
GENPRFM	IP.CMU	IP.PD	IP.PS	MODEL	OS.NAME
GENSTO	IP.CMU	IP.PD	IP.PS	MODEL	OS.NAME
GOTOGEN	IP.CMU	IP.PD	IP.PS	MODEL	OS.NAME
GREPORT	IP.CMU	IP.PD	IP.PS	MODEL	OS.NAME
ITEMCOP	IP.CMU	IP.PD	IP.PS	MODEL	OS.NAME
LEXDATA	IP.CMU	IP.PD	IP.PS	MODEL	OS.NAME
LEXPROC	IP.CMU	IP.PD	IP.PS	MODEL	OS.NAME
LEXY	IP.CMU	IP.PD	IP.PS	MODEL	OS.NAME
LIT02	IP.CMU	IP.PD	IP.PS	MODEL	OS.NAME
MNEMON	IP.CMU	IP.PD	IP.PS	MODEL	OS.NAME
01B	IP.CMU	IP.PD	IP.PS	MODEL	OS.NAME
01R1	IP.CMU	IP.PD	IP.PS	MODEL	OS.NAME
01E	IP.CMU	IP.PD	IP.PS	MODEL	OS.NAME
PASS1B	IP.CMU	IP.PD	IP.PS	MODEL	OS.NAME
PASS1B1	IP.CMU	IP.PD	IP.PS	MODEL	OS.NAME
PASS1C	IP.CMU	IP.PD	IP.PS	MODEL	OS.NAME
PASS1D	IP.CMU	IP.PD	IP.PS	MODEL	OS.NAME
PASS1E	IP.CMU	IP.PD	IP.PS	MODEL	OS.NAME
PASS1F	IP.CMU	IP.PD	IP.PS	MODEL	OS.NAME
PASS1G	IP.CMU	IP.PD	IP.PS	MODEL	OS.NAME
PASS1H	IP.CMU	IP.PD	IP.PS	MODEL	OS.NAME
PASS2	IP.CMU	IP.PD	IP.PS	MODEL	OS.NAME
PRFOPS	IP.CMU	IP.PD	IP.PS	MODEL	OS.NAME
REF	IP.CMU	IP.PD	IP.PS	MODEL	OS.NAME
SAD	IP.CMU	IP.PD	IP.PS	MODEL	OS.NAME
SAVEREG	IP.CMU	IP.PD	IP.PS	MODEL	OS.NAME
SCAN2	IP.CMU	IP.PD	IP.PS	MODEL	OS.NAME
SNAP	IP.CMU	IP.PD	IP.PS	MODEL	OS.NAME
SUBSCR	IP.CMU	IP.PD	IP.PS	MODEL	OS.NAME
TCLIMB	IP.CMU	IP.PD	IP.PS	MODEL	OS.NAME
TROUT1G	IP.CMU	IP.PD	IP.PS	MODEL	OS.NAME

\*\*\*\*SORT\*\*\*\*

\*\*\*\*\*

IPARAMS

\*\*\*\*\*

IP.CMU	SMLOAD	SORTMRG	SMRELOC	DIRECT	MACPRO	CONCRD
	KEYCPL	KEYCMP	KEYCOD	BUFALL	TRNSRT	TSC
	TMC	FMC	FMIP	FMOP	FSRTGET	MRGCON
	MRGDSN	SRTCON	SRTPUT	SRTGET		
IP.CSET	SMRELOC	SMPARAM	MACPRO	CONCRD	ENDPRO	EXTRACT
	SRTMRG	V3CNCRD	KEYCPL			

IPARAMS SORTED BY ROUTINE

BUFALL	IP.CMU	
CONCRD	IP.CMU	IP.CSET
DIRECT	IP.CMU	
ENDPRO	IP.CSET	
EXTRACT	IP.CSET	
FMC	IP.CMU	
FMIP	IP.CMU	
FMOP	IP.CMU	
FSRTGET	IP.CMU	
KEYCMP	IP.CMU	
KEYCOD	IP.CMU	
KEYCPL	IP.CMU	IP.CSET
MACPRO	IP.CMU	IP.CSET
MRGCON	IP.CMU	
MRGDSN	IP.CMU	
SMLOAD	IP.CMU	
SMPARAM	IP.CSET	

SMRELOC	IP.CMU	IP.CSET
SORTMRG	IP.CMU	
SRTCON	IP.CMU	
SRTGET	IP.CMU	
SRTMRG	IP.CSET	
SRTPUT	IP.CMU	
TMC	IP.CMU	
TRNSRT	IP.CMU	
TSC	IP.CMU	
V3CNCRD	IP.CSET	

\*\*\*\*INTERCOM 4\*\*\*\*

\*\*\*\*\*

IPARAMS

\*\*\*\*\*

IP.CMU	STRMOV	KOMSTR				
IP.CPLM	1CI					
IP.C63	ECAFILL					
IP.CSET		9Z1	T76	3TT	1CI	1QP
	1IM	IPP	GBJ	GEJ	GES	1GS
	1GJ	2GJ	1GR	1XP	1LX	1WB
	LCD	0ZZ	1ZZ	0FA	1FE	111
	1BR	1QM	11D	1DS	TBL	FNT
	IUP	IAP	MES	MUJ	MAC	FAD
	DISBEG	STORBEG	FETBEG	REGACT	GETID	IUID
	PAGEDAT	ECAFILL	RESEQ			
IP.IACES	1QP					
IP.ILCMD	1CI	1QP				
IP.INT5		9Z1	T76	3TT	1CI	1QP
	1IM	IPP	GBJ	GEJ	GES	1GS
	1GJ	2GJ	1GR	1XP	1LX	1WB
	LCD	0ZZ	1ZZ	0FA	1FE	111
	1BR	1QM	11D	1DS	TBL	FNT
	IUP	IAP	MES	MUJ	MAC	FAD
	DISBEG	STORBEG	FETBEG	REGACT	GETID	IUID
IP.IUSID	111	SETIPS				
IP.MPRIT	1CI	1XP	1LX			
IP.IPRLS	1CI	1LX				
IP.1WB	1CI					
OS.NAME	RESEQ					

\*\*\*\*\*

ECSCOM

\*\*\*\*\*

IPARAMS SORTED BY ROUTINE

DISBEG	IP.CSET	IP.INT5
ECAFILL	IP.CSET	IP.INT5
FAD	IP.C63	IP.CSET
FETBEG	IP.CSET	IP.INT5
FNT	IP.CSET	IP.INT5
GBJ	IP.CSET	IP.INT5
GEJ	IP.CSET	IP.INT5
GES	IP.CSET	IP.INT5
GETID	IP.CSET	IP.INT5
IAP	IP.CSET	IP.INT5
IPP	IP.CSET	IP.INT5
IUID	IP.CSET	IP.INT5
IUP	IP.CSET	IP.INT5
KOMSTR	IP.CMU	
LCD	IP.CSET	IP.INT5
MAC	IP.CSET	IP.INT5
MES	IP.CSET	IP.INT5
MUJ	IP.CSET	IP.INT5
PAGEDAT	IP.CSET	IP.INT5
REGACT	IP.CSET	IP.INT5
RESEQ	IP.CSET	OS.NAME
SETIPS	IP.IUSID	
STORBEG	IP.CSET	IP.INT5
STRMOV	IP.CMU	
TBL	IP.CSET	IP.INT5
T76	IP.CSET	IP.INT5



OFA	IP.CSET	IP.INT5				
OZZ	IP.CSET	IP.INT5				
1BR	IP.CSET	IP.INT5				
1CI	IP.CPLM	IP.CSET	IP.ILCMD	IP.INT5	IP.MPRIT	IP.IPRLS
	IP.1WB					
1DS	IP.CSET	IP.INT5				
1FE	IP.CSET	IP.INT5				
1GJ	IP.CSET	IP.INT5				
1GR	IP.CSET	IP.INT5				
1GS	IP.CSET	IP.INT5				
1ID	IP.CSET	IP.INT5				
1IM	IP.CSET	IP.INT5				
1I1	IP.CSET	IP.INT5	IP.IUSID			
1LX	IP.CSET	IP.INT5	IP.MPRIT	IP.IPRLS		
1QM	IP.CSET	IP.INT5				
1QP	IP.CSET	IP.IACES	IP.ILCMD	IP.INT5		
1WB	IP.CSET	IP.INT5				
1XP	IP.CSET	IP.INT5	IP.MPRIT			
1ZZ	IP.CSET	IP.INT5				
2GJ	IP.CSET	IP.INT5				
3TT	IP.CSET	IP.INT5				
9Z1	IP.CSET	IP.INT5				

\*\*\*\*INTERCOM 5\*\*\*\*

\*\*\*\*\*

IPARAMS

\*\*\*\*\*

IP.CMU	STRMOV	KOMSTR				
IP.CPLM	1CI					
IP.C63	ECAFILL					
IP.CSET	3TT	1CI	1IM	1ID	1DS	MES
	MAC	PAGEDAT	ECAFILL	RESEQ		
IP.IACES	1QP					
IP.ILCMD	1CI	1QP				
IP.INT5	T76	3TT	1CI	1QP	1IM	IPP
	LCD	1NI	1DI	1ND	1I1	1NP
	1BR	1QM	1ID	1DS	TBL	FNT
	IUP	IAP	MES	MUJ	MAC	FAD
	DISBEG	STORBEG	FETBEG	REQACT	SCREEN	GETID
	IUID	INTRST				
IP.IUSID	1I1	SETIPS				
IP.MPRIT	1CI	1NP				
IP.IPRLS	1CI	1NP				
IP.X780	T76	3TT	1CI	1QP	1IM	IPP
	LCD	1NI	1DI	1ND	1I1	1NP
	1BR	1QM	1ID	1DS	TBL	FNT
	IUP	IAP	MES	MUJ	MAC	FAD
	DISBEG	STORBEG	FETBEG	REQACT	SCREEN	GETID
	IUID	INTRST				
	RESEQ					

OS.NAME

\*\*\*\*\*

ECSCOM

\*\*\*\*\*

IPARAMS SORTED BY ROUTINE

DISBEG	IP.INT5	IP.X780	
ECAFILL	IP.C63	IP.CSET	
FAD	IP.INT5	IP.X780	
FETBEG	IP.INT5	IP.X780	
FNT	IP.INT5	IP.X780	
GETID	IP.INT5	IP.X780	
IAP	IP.INT5	IP.X780	
INTRST	IP.INT5	IP.X780	
IPP	IP.INT5	IP.X780	
IUID	IP.INT5	IP.X780	
IUP	IP.INT5	IP.X780	
KOMSTR	IP.CMU		
LCD	IP.INT5	IP.X780	
MAC	IP.CSET	IP.INT5	IP.X780
MES	IP.CSET	IP.INT5	IP.X780
MUJ	IP.INT5	IP.X780	
PAGEDAT	IP.CSET		

REQACT	IP.INT5	IP.X780				
RESEQ	IP.CSET	OS.NAME				
SCREEN	IP.INT5	IP.X780				
SETIPS	IP.IUSID					
STORBEG	IP.INT5	IP.X780				
STRMOV	IP.CMU					
TBL	IP.INT5	IP.X780				
T76	IP.INT5	IP.X780				
1BR	IP.INT5	IP.X780				
1CI	IP.CPLM	IP.CSET	IP.ILCMD	IP.INT5	IP.MPRIT	IP.IPRLS
	IP.X780					
1DI	IP.INT5	IP.X780				
1DS	IP.CSET	IP.INT5	IP.X780			
1ID	IP.CSET	IP.INT5	IP.X780			
1IM	IP.CSET	IP.INT5	IP.X780			
1II	IP.INT5	IP.IUSID	IP.X780			
1ND	IP.INT5	IP.X780				
1NI	IP.INT5	IP.X780				
1NP	IP.INT5	IP.MPRIT	IP.IPRLS	IP.X780		
1QM	IP.INT5	IP.X780				
1QP	IP.IACES	IP.ILCMD	IP.INT5	IP.X780		
3TT	IP.CSET	IP.INT5	IP.X780			

\*\*\*\*ALGOL 4 (ALGEDIT)\*\*\*\*

\*\*\*\*\*  
IPARAMS  
\*\*\*\*\*

IP.C63

SQIOWA

IPARAMS SORTED BY ROUTINE

SQIOWA

IP.C63

\*\*\*\*BASIC 3\*\*\*\*

\*\*\*\*\*  
IPARAMS  
\*\*\*\*\*

IP.CSET  
IP.PD  
IP.PS  
OS.NAME

BASRTS  
BASCARD  
BASCARD  
BASOGEN  
BASIINP

BASTRNG  
BASEGEN  
BASEGEN  
BASTRNG

BASPRUS  
BASPRUS  
BASPRUS

BASCOMP  
BASERRS  
BASCHAN

BASSINT  
BASCOMP

BASIGEN  
BASOPTS

IPARAMS SORTED BY ROUTINE

BASCARD  
BASCHAN  
BASCOMP  
BASEGEN  
BASERRS  
BASIGEN  
BASIINP  
BASOGEN  
BASOPTS  
BASPRUS  
BASRTS  
BASSINT  
BASTRNG

IP.PD  
OS.NAME  
IP.CSET  
IP.PD  
OS.NAME  
OS.NAME  
OS.NAME  
OS.NAME  
OS.NAME  
OS.NAME  
IP.CSET  
IP.CSET  
OS.NAME  
IP.CSET

IP.PS  
OS.NAME  
OS.NAME  
OS.NAME  
OS.NAME  
OS.NAME  
OS.NAME  
OS.NAME  
OS.NAME  
OS.NAME  
OS.NAME

\*\*\*\*DBU\*\*\*\*

\*\*\*\*\*  
IPARAMS  
\*\*\*\*\*

OS.NAME

DF\$EXT

DF\$EXTS

DF\$LCAL

\*\*\*\*\*  
CIOCOM  
\*\*\*\*\*  
\*\*\*\*\*  
ECSCOM

\*\*\*\*\*

IPARAMS SORTED BY ROUTINE

DFSEXT	OS.NAME
DFSEXTS	OS.NAME
DF\$LCAL	OS.NAME

\*\*\*\*COBOL 5\*\*\*\*

\*\*\*\*\*

IPARAMS

\*\*\*\*\*

IP.CMU	CB5TEXT
IP.YMD	CSACCDT
MODEL	CB5TEXT

\*\*\*\*\*

CIOCOM

\*\*\*\*\*

\*\*\*\*\*

ECSCOM

\*\*\*\*\*

IPARAMS SORTED BY ROUTINE

CSACCDT	IP.YMD	
CB5TEXT	IP.CMU	MODEL

\*\*\*\* FORTRAN 5 \*\*\*\*

\*\*\*\*\*

IPARAMS

\*\*\*\*\*

IP.CSET	LEX					
IP.MFL	FTN					
IP.PD	FTN5TXT	FTN	PUC	INIT00	INIT20	FTN510
	FTN520					
IP.PS	FTN5TXT	INIT00				
MODEL	FTN5TXT	FTN	PUC	FEC	FSNAP	LEX
	HEADER	KEY	DATA	DECL	TYPE	PAR
	CONRED	LABEL	QCGC	FUN	REG	GEN
	REC	FAS	MAP	LIST	CCGC	BRIDGE
OS.NAME	FTN5TXT					
OS.ID	FAS					
OS.VER	FTN5TXT					

IPARAMS SORTED BY ROUTINE

BRIDGE	MODEL					
CCGC	MODEL					
CONRED	MODEL					
DATA	MODEL					
DECL	MODEL					
FAS	MODEL	OS.ID				
FEC	MODEL					
FSNAP	MODEL					
FTN	IP.MFL	IP.PD	MODEL			
FTN5TXT	IP.PD	IP.PS	MODEL	OS.NAME	OS.VER	
FTN510	IP.PD					
FTN520	IP.PD					
FUN	MODEL					
GEN	MODEL					
HEADER	MODEL					
INIT00	IP.PD	IP.PS				
INIT20	IP.PD					
KEY	MODEL					
LABEL	MODEL					
LEX	IP.CSET	MODEL				
LIST	MODEL					
MAP	MODEL					
PAR	MODEL					
PUC	IP.PD	MODEL				
QCGC	MODEL					
REC	MODEL					
REG	MODEL					
TYPE	MODEL					

\*\*\*\* FTN 5 LIBRARY \*\*\*\*

\*\*\*\*\*  
 IPARAMS  
 \*\*\*\*\*

IP.CSET	Q5NTRY= OUTC=	COMIO= OUTF=	ENCODE= COLSEQ=	INCOM= FERCAP=	KRAKER=	LDIN=
MODEL	Q5NTRY= OIFC=	DECODE= MOVLEV	ENCODE= READEC	FORSYS= WRITEC	IDAC= FERCAP=	IIFC= RPVCAP=
OS.NAME	MATHTEX Q5NTRY=	XTOY. CONDIS=	FORSYS=	LDOUT=	NAMOUT=	OUTF=
OS.VER	PAUSE= Q5NTRY=	RPVCAP= FORSYS=	MATHTEX MATHTEX			

IPARAMS SORTED BY ROUTINE

COLSEQ=	IP.CSET				
COMIO=	IP.CSET				
CONDIS=	OS.NAME				
DECODE=	MODEL				
ENCODE=	IP.CSET	MODEL			
FERCAP=	IP.CSET	MODEL			
FORSYS=	MODEL	OS.NAME	OS.VER		
IDAC=	MODEL				
IIFC=	MODEL				
INCOM=	IP.CSET				
KRAKER=	IP.CSET				
LDIN=	IP.CSET				
LDOUT=	OS.NAME				
MATHTEX	MODEL	OS.NAME	OS.VER		
MOVLEV	MODEL				
NAMOUT=	OS.NAME				
OIFC=	MODEL				
OUTC=	IP.CSET				
OUTF=	IP.CSET	OS.NAME			
PAUSE=	OS.NAME				
Q5NTRY=	IP.CSET	MODEL	OS.NAME	OS.VER	
READEC	MODEL				
RPVCAP=	MODEL	OS.NAME			
WRITEC	MODEL				
XTOY.	MODEL				

\*\*\*\* FORTRAN DATA BASE FACILITY \*\*\*\*

\*\*\*\*\*  
 IPARAMS  
 \*\*\*\*\*

IP.PD	ARG
OS.NAME	CBIO
OS.ID	PRFX

IPARAMS SORTED BY ROUTINE

ARG	IP.PD
CBIO	OS.NAME
PRFX	OS.ID

\*\*\*\* ALGOL 5 \*\*\*\*

\*\*\*\*\*  
 IPARAMS  
 \*\*\*\*\*

IP.PD	A60TEXT	PTLST	CONCARD	CONCRD0
IP.PS	A60TEXT	CONCARD	CONCRD0	
MODEL	A60TEXT	A\$OUTPT		
OS.NAME	A60TEXT	MAIN70	INIT60	
OS.ID	MAIN70	INIT60		
OS.VER	A60TEXT			

IPARAMS SORTED BY ROUTINE

A\$OUTPT	MODEL				
A60TEXT	IP.PD	IP.PS	MODEL	OS.NAME	OS.VER
CONCARD	IP.PD	IP.PS			

CONCRD0	IP.PD	IP.PS
INIT60	OS.NAME	OS.ID
MAIN70	OS.NAME	OS.ID
PTLST	IP.PD	

\*\*\*\* DATA CATALOGUE \*\*\*\*

\*\*\*\*\*  
IPARAMS  
\*\*\*\*\*

MODEL	FICATION		
	IPARAMS SORTED BY	ROUTINE	
FICATION	MODEL		

\*\*\*\*CDCS 2\*\*\*\*

\*\*\*\*\*  
IPARAMS  
\*\*\*\*\*

OS.NAME	DBQRFI DB\$IO	DBSATCM	DB\$ATCR	DB\$CPT	DB\$DMGI	DB\$EXT
	IPARAMS SORTED BY	ROUTINE				
DB\$ATCM	OS.NAME					
DB\$ATCR	OS.NAME					
DB\$CPT	OS.NAME					
DB\$DMGI	OS.NAME					
DB\$EXT	OS.NAME					
DB\$IO	OS.NAME					
DBQRFI	OS.NAME					

\*\*\*\*PL/I\*\*\*\*

\*\*\*\*\*  
IPARAMS  
\*\*\*\*\*

IP.C63	RTSTEXT		
IP.CSET	PLITEXT	RTSTEXT	
IP.PD	PLI	PLI10	
IP.PS	PLI		
IP.YMD	IOBILT?		
MODEL	PLI	ZZZZPLI	
OS.NAME	RTSTEXT		
	IPARAMS SORTED BY	ROUTINE	
IOBILT?	IP.YMD		
PLI	IP.PD	IP.PS	MODEL
PLITEXT	IP.CSET		
PLI10	IP.PD		
RTSTEXT	IP.C63	IP.CSET	OS.NAME
ZZZZPLI	MODEL		

\*\*\*\*CYBER INTERACTIVE DEBUG\*\*\*\*

\*\*\*\*\*  
IPARAMS  
\*\*\*\*\*

IP.CSET	ASCORD				
MODEL	INTERP				
OS.NAME	DBGRPV	DEBUG.	DEBUGOM	GETPOS	GETSTAT
	MINIGO	MININI	MINSYS	RTNALL	SETRPV
	SWAPMI	SWAPUI	SWAPUO	WEBUG	DEBUG
OS.ID	MAXLDR				MAXLDR
					SPSBRT
	IPARAMS SORTED BY	ROUTINE			
ASCORD	IP.CSET				
DBGRPV	OS.NAME				
DEBUG.	OS.NAME				

DEBUGOM	OS.NAME	
DEBUG	OS.NAME	
GETPOS	OS.NAME	
GETSTAT	OS.NAME	
INTERP	MODEL	
MAXLDR	OS.NAME	OS.ID
MINIGQ	OS.NAME	
MININI	OS.NAME	
MINSYS	OS.NAME	
RTNALL	OS.NAME	
SETRPV	OS.NAME	
SPSBRT	OS.NAME	
SWAPMI	OS.NAME	
SWAPUI	OS.NAME	
SWAPUO	OS.NAME	
WEBUG	OS.NAME	
KIDIEAI	IP.LDS	

\*\*\*\*COMMON CODE GENERATOR\*\*\*\*

\*\*\*\*\*  
 IPARAMS  
 \*\*\*\*\*

MODEL	GPO
OS.ID	CGIA

IPARAMS SORTED BY ROUTINE

CGIA	OS.ID
GPO	MODEL

# INDEX

- AAM II-4-1
- Abnormal termination I-1-8.1
- ACCOUNT I-1-8.1
- ACCTVAL micro II-1-51
- ADD statement II-12-26; II-28-24
- Advanced access methods II-4-1
- AK parameters II-4-3
- ALGOL 4 I-1-12; II-13-1
- ALGOL 5 I-1-12; II-36-1
- ALGTEXT I-1-12
- ALG5TEXT I-1-12
- Allocation style II-28-5
- ASYN macro II-28-8
- AUDITKEY II-1-45
- AUTO macro II-27-1
- AUTOMAT procedure I-1-6
- Automatic assignment bit II-1-13
- Autopmatic scratch status II-1-13
  
- BAM II-3-1
- BASE I-1-7
- BASIC II-21-1
- Basic access methods II-3-1
- BISY NC macro II-28-8,9
- Build techniques I-1-2
  
- CATALOG I-1-8.1
- CCI II-29-1
- CCL II-26-1
- CCP I-1-8.1; II-23-1
- CCP PASCAL II-15-2
- CCP64 I-1-8.1
- CDCS 1 I-1-8.1; II-18-1
- CDCS 2 II-30-1
- CE diagnostics II-7-1
- Central exchange jump switch II-1-2
- Change orders II-1-2
- Character sets I-1-8.1
- CHAR64 II-1-45
- CIA II-8-2
- CID II-32-1
- CIOCOM II-1-10
- CIOCP I-1-8.1
- CMM II-1-45
- CMR I-1-8,11; II-1-10
  - Configuration parameters II-1-16; II-12-8; II-28-7
  - Equipment configuration examples II-1-22,27
  - Multiplexer table II-12-8; II-28-7; II-40-2
- CMRLIB I-1-8
- CMRS I-1-9
- CMS test I-1-8.1
- CMU I-1-8.1
- COBOL 4 II-10-1
- COBOL 4 to COBOL 5 conversion aid II-25-1
- COBOL 5 I-1-8.1; II-24-1
- Coldstart deadstart II-1-2
- Command table structure (1CI) II-12-13; II-28-10
- Common decks I-1-2,9,11,12
- Common common decks II-2-1
- Common LIDs II-1-56
- Coldstart deck re-creation II-1-46
- Common memory manager II-1-45
- Communication control INTERCOM II-29-1
- Communications control program II-23-1
- COMPASS II-2-1
- COMPCOM I-1-11
- COMP3\$ II-2-1
- Computing physical addresses II-1-26
- COMTBL deck II-28-10
- COM1LX macro II-12-14
- COM2CC macro II-12-13; II-28-10
- Configuration, equipment II-1-20
- Controller type II-1-21
- Controlware binary creation II-1-46
- Conversion aids II-8-1; II-25-1; II-37-1
- COPBC II-1-47
- COPY8P II-5-1
- CPMET II-8-2
- CREATE II-4-1
- CRM Refer to CDC CYBER record manager
- Cross-reference lists III-1-1
- CWARE I-1-7,8,9
- CDC CYBER control language II-26-1
- CDC CYBER cross system II-15-1
- CDC CYBER database control system 1 II-18-1
- CDC CYBER database control system 2 II-30-1
- CDC CYBER interactive debug II-32-1
- CDC CYBER LOADER II-16-1
- CDC CYBER record manager I-1-11,12; II-3-1; II-4-1
- CDC CYBER utilities II-1-56
- CPCTEXT I-1-11
- CP.OVLIB II-2-1
- CT3 II-1-2
  
- DA parameters II-4-3
- Data base utilities II-22-1
- Data Catalogue installation II-41-1
- Data description language 2 II-20-1
- Data description language 3 II-31-1
- DBU II-22-1
- DDL 2 II-20-1
- DDL 3 II-31-1
- Deadstart diagnostic sequencer I-1-2,3
  - Installation parameters II-1-29
  - Loading II-1-46
  - Tape contents, unconfigured II-1-2
  - Tape creation I-1-2
  - Tape model job II-1-48
- Deadstart dump analyzer II-1-58
- Deck options I-1-7
- DECLKEY II-1-45
- DEL statement II-12-27; II-28-24
- Dependencies, product I-1-4
- Device set installation II-1-31
- Device type II-1-21
- DIM I-1-8.1
- Disk space threshold II-1-33
- DL2 I-1-8.1
- DMGMNT I-1-8; II-19-1
- DOCK II-8-1,2
- Downline load modules II-23-1,6; II-29-1

DPFC macro II-1-15  
Driver type selection II-7-1; II-12-12  
DST1 I-1-2,9,10; II-1-49  
DST2 I-1-2,9,10; II-1-49  
DST3 I-1-11; II-1-49  
DYNAMFL II-1-45  
D44 tests II-17-1

ECS I-1-8  
  Installation parameters II-1-41  
  Label II-1-42  
  Link II-1-21,44  
  Segment activity count II-1-43  
  System segments II-1-43  
  Usage console entries II-1-40  
ECSBUF macro II-1-42  
ECSLABEL macro II-1-42.1  
ECSPART macro II-1-42.1  
EDITKEY II-1-45  
EDITLIB I-1-3  
EDITOR  
  Array dimensions in IPFTN II-12-21; II-28-20  
  DEBUG code II-12-22; II-28-20  
  Dependency chart II-12-22; II-28-19  
  Installation parameters II-12-17,27; II-28-14,16  
Empty common decks I-1-9  
EMPTY macro II-28-8  
Enhanced station support I-1-8.1; II-1-49  
Equipment configuration II-1-20  
EST II-1-20; II-12-7; II-40-2  
ESTMATE II-4-1  
ES4IMS I-1-8.1  
ES4XREF I-1-8.1  
EXIT I-1-8.1  
EXPCOM macro II-28-10  
EXPORT HS installation II-40-1  
EXTOVLP II-1-45

Factory format support (844) II-17-1  
FCL II-34-1  
FCOs II-1-2  
FDBF II-35-1  
FLAW table II-1-25  
Flaws II-17-1  
FORM II-6-1  
FORMAT utility II-17-2  
FORMAT/FDP II-17-1  
FORTRAN common library II-34-1; II-39-1  
FORTRAN Data Base Facility II-35-1  
FORTRAN Extended 4 I-1-8,12; II-9-1  
FORTRAN 4/5 conversion aid II-37-1  
FORTRAN 5 installation II-38-1  
FTNMAC I-1-12  
FTNTEXT II-9-2

Gemini installation II-1-56  
GENTBL macro II-4-4  
Graphics II-12-5

Hardware feature micro II-1-5  
Hard-wired terminals II-12-10  
HD I-1-7,8  
HELLO7 I-1-8.1  
HF.LIST micro II-1-5  
HY I-1-7,8

ID table II-1-18

Installation

Decks I-1-1  
Job extraction I-1-6,7  
Order I-1-4

Parameters (IP.symbols)

IP.ACNT II-1-6  
IP.ARCH II-1-6  
IP.AS II-21-1  
IP.ATT II-26-2  
IP.BL II-21-1  
IP.BSIM II-28-5  
IP.BUFFE II-28-6  
IP.CBKSP II-1-10  
IP.CMPID II-1-43  
IP.CMU II-1-6  
IP.CP II-1-6  
IP.CPLM II-1-6.1  
IP.CR II-1-6.1  
IP.CSET II-1-6.1; II-21-1  
IP.CTCT II-12-12  
IP.CTCT4 II-12-12  
IP.CTCT9 II-12-12  
IP.CYSTP II-1-44  
IP.C176 II-1-6.1  
IP.DBAL II-1-6.1  
IP.DCT II-1-17  
IP.DSRMS II-1-6.1  
IP.DPF II-26-1  
IP.DPFN II-26-1  
IP.D7DN II-1-10  
IP.D9DN II-1-11  
IP.EBUF II-1-41  
IP.ECLNK II-1-44  
IP.ECSB II-1-6.1,44  
IP.ECSLK II-1-44  
IP.ECSTP II-1-17  
IP.ECSW II-1-42  
IP.ECYC II-1-44  
IP.EIDLE II-1-44  
IP.ELIB II-1-41  
IP.ELST II-1-18  
IP.ERES II-1-41  
IP.EXP II-26-2  
IP.FLINC II-16-2  
IP.FLMSG II-16-2  
IP.FP II-26-1  
IP.FPC II-26-1  
IP.FTHRL II-1-17  
IP.FTHRU II-1-17  
IP.FTNTS II-28-3  
IP.IACES II-1-6.1; II-12-3,7; II-28-3,6  
IP.IBBMN II-12-5  
IP.IBBMX II-12-5  
IP.ID II-12-5; II-26-2; II-28-4  
IP.IDFL II-12-5; II-28-4  
IP.IDFLE II-12-5; II-28-4  
IP.IFC II-12-5  
IP.IGCON II-12-5  
IP.IGS II-12-5  
IP.IHEAD II-12-5; II-28-4  
IP.IHSBS II-28-5  
IP.IHSLW II-28-5  
IP.IIBMN II-12-5; II-28-4  
IP.IIBMX II-12-5; II-28-4  
IP.ILCMD II-12-7; II-28-6  
IP.IMXL3 II-12-6  
IP.IMXL4 II-12-6  
IP.IM2DS II-12-5  
IP.IM2DW II-12-5  
IP.IM3BS II-28-4  
IP.IM3DS II-12-5  
IP.IM3DW II-12-5



IP.IM3LW II-28-5  
 IP.IM4DS II-12-5  
 IP.IM4DW II-12-5  
 IP.IND II-12-6; II-28-5  
 IP.INT5 II-28-6  
 IP.IPRLS II-12-6; II-28-5  
 IP.IOLM II-1-6.1  
 IP.IQD II-1-7  
 IP.IQPW II-1-7  
 IP.ISFL II-12-6; II-28-5  
 IP.IUSID II-12-7; II-28-6  
 IP.IWT II-12-6  
 IP.I27BS II-28-5  
 IP.I27LW II-28-5  
 IP.I37BS II-28-5  
 IP.I37LW II-28-5  
 IP.I4ABS II-28-5  
 IP.I4ALW II-28-5  
 IP.I4CBS II-28-5  
 IP.I4CLW II-28-5  
 IP.LCS II-26-1  
 IP.LDBG II-16-2  
 IP.LDER II-16-2  
 IP.LINK II-1-7,61  
 IP.LNKBF II-1-44  
 IP.LNKMN II-1-44  
 IP.LP4C II-12-6  
 IP.LRT II-16-2  
 IP.LVF II-1-7  
 IP.MALOC II-12-6; II-28-5  
 IP.MAP II-16-2  
 IP.MAXBL II-1-44  
 IP.MCPU II-1-7  
 IP.MECS II-1-7  
 IP.MFL II-1-7  
 IP.MMS II-1-7  
 IP.MPPU II-1-7  
 IP.MPR II-1-7  
 IP.MPRIT II-12-6; II-28-5  
 IP.MSCT II-1-8  
 IP.MSLM II-1-8  
 IP.MTL II-1-8  
 IP.MXCOR II-12-6; II-28-5  
 IP.NBCD II-1-11  
 IP.NBRK II-1-11  
 IP.NDEN II-1-11  
 IP.NDFS II-1-8  
 IP.NTCN II-1-11  
 IP.NOISE II-1-11  
 IP.NOIS9 II-1-11  
 IP.OPRI II-1-8  
 IP.OQD II-1-8  
 IP.PD II-1-8,56; II-16-2; II-21-1  
 IP.PFRP II-1-8  
 IP.PNL II-26-1  
 IP.POSFL II-1-8  
 IP.PPS2X II-1-8; II-12-3  
 IP.PRIX II-12-6; II-28-5  
 IP.PRUB II-28-6  
 IP.PS II-1-8; II-16-2; II-21-1  
 IP.PSET II-16-1  
 IP.PTCN II-1-11  
 IP.RCYC II-1-11  
 IP.REW II-16-2  
 IP.RM II-1-8  
 IP.RPE1 II-1-11  
 IP.RPE2 II-1-11  
 IP.SCHDE II-1-8.1  
 IP.SCS II-26-1  
 IP.SECS II-1-9  
 IP.SFL II-1-9  
 IP.SIDLE II-1-9

IP.SIOL II-1-9  
 IP.SLIB II-1-18  
 IP.SMS II-1-9  
 IP.SNIFL II-12-6  
 IP.SNOFL II-12-6  
 IP.SNOTL II-12-6  
 IP.SPR II-1-9  
 IP.SPT II-1-9  
 IP.SRMS I-1-8.1; II-1-9  
 IP.STEX II-1-55  
 IP.STL II-1-9  
 IP.SYS II-26-2  
 IP.SYSE II-1-18  
 IP.SYSL1 II-1-18  
 IP.TCPUB II-1-9  
 IP.TDEN II-1-11  
 IP.TSL II-12-7; II-28-5  
 IP.TYPE II-1-9  
 IP.UP II-1-9  
 IP.US II-1-10  
 IP.VER II-1-18; II-26-2  
 IP.WEC II-1-15  
 IP.XJ II-1-10  
 IP.X780 II-28-6  
 IP.YMD II-1-10  
 IP.1LX II-12-7  
 IP.1M1 II-12-7  
 IP.1WB II-12-7  
 IP.1ZZ II-12-7  
 IP.7LNK II-1-55  
 IP.819 II-1-10  
 Procedure, CDC CYBER 171 configuration II-27-1  
 Requirements I-1-5  
 Sequence I-1-6  
 Integer multiply II-1-2  
 INTERCOM 4 I-1-8.1  
 INTERCOM 5 II-28-1  
 IOTEXT I-1-11  
 IPCOM II-12-17; II-28-14  
 IPFILL II-12-17; II-28-14  
 IPFTN II-12-17; II-28-14  
 IPTXT I-1-11; II-7-1  
 IS parameters II-4-2  
 IXGEN II-4-1  
  
 Job Clases II-1-35  
  
 KYAN II-4-1  
  
 LCC programs II-12-23  
 LIBS I-1-10  
 Line speeds II-12-10; II-28-8  
 LINP II-16-2  
 LIST I-1-8  
 LOADER II-16-1  
 Load leveling II-1-54,55,56  
 LOCALCC I-1-9  
 LOCALIN I-1-9  
 LOCLIB I-1-8  
 Logical ID table II-1-18.1  
 L.ITABL II-12-8; II-28-7  
  
 Maintenance package II-8-1  
 Maintenance tools II-8-1

Master device II-1-21  
MESSAGE processing II-1-2  
MINI I-1-10  
Miscellaneous bits II-1-14  
MODEL II-1-10  
Model job create deadstart tape II-1-48  
MODE4 macro II-28-2  
MODLEVEL micro II-21-1  
MORENUC I-1-9  
MOREOV I-1-9  
MORESYS I-1-9  
MOREPP I-1-9  
MOREPP I-1-9  
MTS I-1-8; II-1-21  
MUJ  
  IPARAMS II-12-22; II-28-20  
  Macro II-12-14; II-28-10  
  System errors II-12-24; II-28-22  
  Table II-12-13; II-28-12  
MUJTBL macro II-28-12  
Multiplexer definition entries II-12-9; II-28-7,8  
Multidrop lines II-12-10  
Multistation terminals II-12-11  
MUXCR macro II-27-2  
MUX2550 macro II-28-8

NO1M1 I-1-8.1  
Noise bracketing II-1-11  
NOISEL II-7-1  
Nomenclature, installation deck I-1-4  
NOS/BE 1 II-1-1  
NPU  
  Configuration aids II-29-7  
  Programs II-12-24; II-28-21

Obtaining installation jobs I-1-6  
OLMS II-7-1  
On-line maintenance software II-7-1  
OPTFTN I-1-8  
Option dependencies II-1-14  
OPTION macro II-1-16  
Order of installation I-1-4  
OS.ID II-1-10  
OSMINIT II-29-2  
Overcommitment bit II-1-14

Parity retries II-1-11  
PASCAL I-1-8.1; II-15-1,2  
Password file creation II-12-25; II-28-23  
Password, system dayfile II-1-15  
Password, universal II-1-33  
PASSWRD utility II-12-26; II-28-23  
PE I-1-7,8  
Permanent file installation II-1-31  
PFC II-1-15  
PL naming conventions I-1-4  
PL/I II-33-1  
PL1AT I-1-11  
PMDKEY II-1-45  
Polling times II-1-52  
Port definition entries II-12-9; II-28-8  
Port distributions II-12-11  
Post mortem dump II-34-1; II-39-1  
PPTXT II-1-2  
Prestaging bits II-1-13  
Priority, INTERCOM output file II-12-6; II-28-5  
Programmable format control II-1-15

Q.ILNOFC II-28-6  
QUERY UPDATE II-19-1

RBR II-1-24  
READS I-1-9  
REASON I-1-1,7  
Record block reservation table II-1-24  
REMOTE macro II-12-14; II-28-10  
Retention cycle II-1-11  
RMS deadstart II-1-6.1

SAVE I-1-8.1  
SAVEPL I-1-9  
SCED installation parameters II-12-27; II-28-24  
SCED macro II-12-27; II-28-24  
Scheduling parameters II-1-35,37  
SDFRD micro II-1-15  
SDFXR micro II-1-15  
Segment activity count II-1-43  
SIFT II-8-1  
SIP II-8-5  
SISTAT II-4-1  
SIS63 II-8-2,8  
SMTEXT I-1-12  
Sort/merge I-1-12; II-1-1  
Spacing code arrays II-1-15  
SPOTs II-1-49  
SPOTs, accounting II-1-51  
SPY II-8-1  
SRMS I-1-8.1  
SRMSTXT I-1-12  
Station buffer sizes II-1-53  
Station support II-1-49  
STIMULATOR II-8-1,3  
Symmetric/replacement station II-1-51  
SYMPL I-1-8.1; II-8-1  
SYNGEN II-19-1; II-20-1; II-31-1  
SYSPROC I-1-9  
System  
  Circular buffers (SCB) II-1-42  
  Dayfile password II-1-15  
  Idle mode II-1-40  
  Residency II-1-49  
  Seconds II-1-18.1  
  Texts I-1-11,12

Tape creation, deadstart I-1-2  
  Density I-1-7,8; II-1-11  
  Processing parameters II-1-10  
  Scheduling notes II-1-3,4  
  Scheduling parameters II-1-12  
Tape scheduling options II-1-12  
TBL assembly options II-12-15; II-28-12  
Terminals II-12-1  
Threshold, disk space II-1-33  
Time-out II-12-6  
TSFTN I-1-8  
TSTEXT II-9-2  
TXTREC I-1-9  
Type 1 statements II-29-5  
Type 2 statements II-29-6  
Type 3 statements II-29-6

ULIB I-1-2,8.1,10  
UPDATE installation options II-1-45

Verification jobs I-1-4

Weighting constants, system seconds II-1-19

XTRALIBS I-1-9

0F3 I-1-8.1  
0F4 I-1-8.1

0F7 I-1-8.1  
0MT I-1-8; II-1-46  
0SJ I-1-8; II-1-46  
0SY I-1-8; II-1-46  
0SZ I-1-8; II-1-46  
1QP II-28-12,13  
2CS II-28-12,13  
63CSET I-1-8  
66x tape subsystem I-1-8  
8-bit subroutines II-5-1  
819 RMS I-1-11; II-1-2  
844 RMS II-1-21  
844 full-tracking controlware I-1-8



# COMMENT SHEET

MANUAL TITLE: CDC NOS/BE Version 1 Installation Handbook

PUBLICATION NO.: 60494300

REVISION: M

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).

CUT ALONG LINE

AA3419 REV. 4/79 PRINTED IN U.S.A.

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

FOLD ON DOTTED LINES AND STAPLE

STAPLE

STAPLE

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



CUT ALONG LINE

FOLD

FOLD