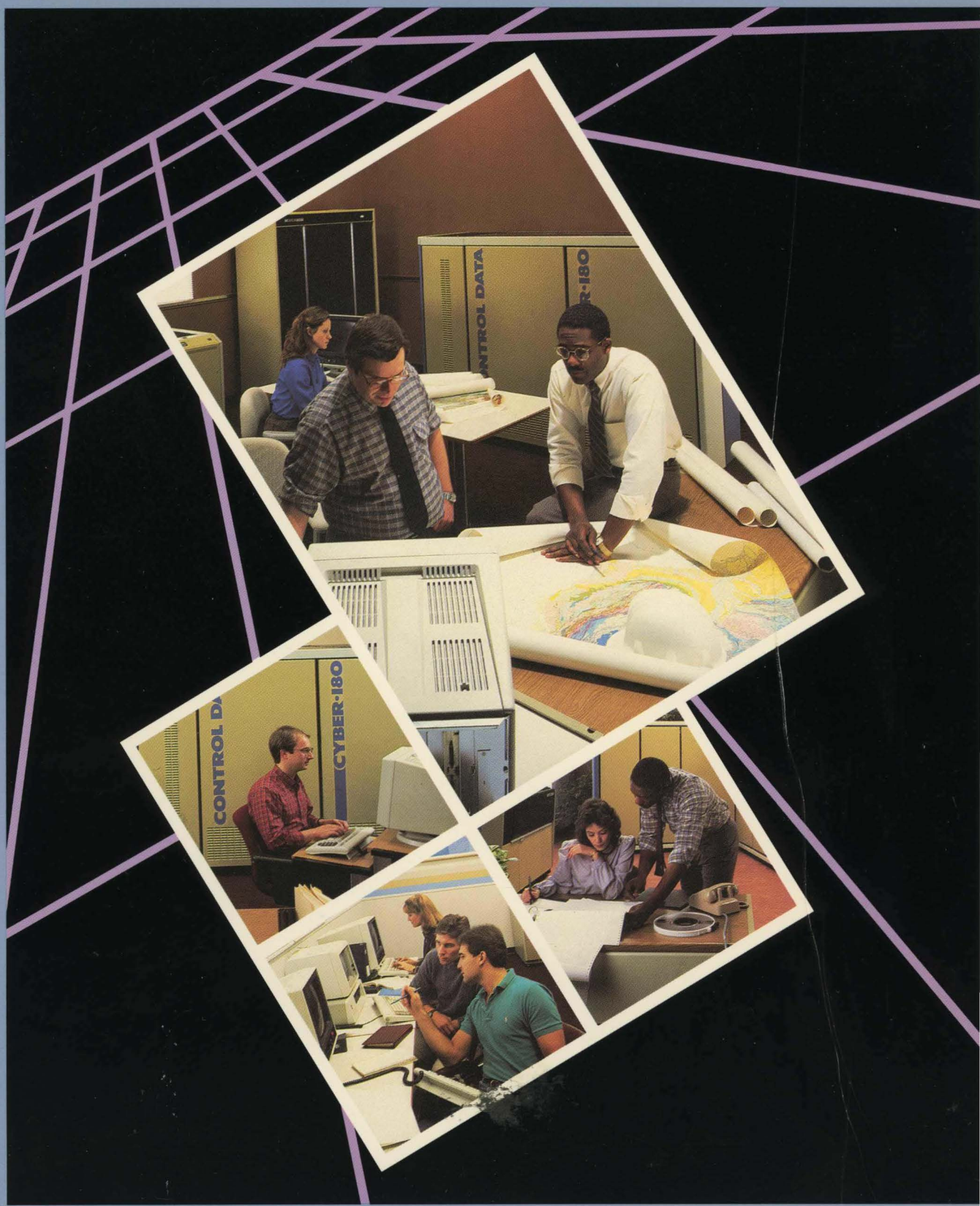
NOS 2 Reference Set Volume 4
Program Interface

CONTROL DATA



60459690

CONTROL DATA

NOS VERSION 2
REFERENCE SET

VOLUME 4

PROGRAM INTERFACE

CDC® COMPUTER SYSTEMS:
CYBER 180
CYBER 170
CYBER 70
6000 SERIES

REVISION RECORD

REVISION	DESCRIPTION
A (04-26-82)	Manual released. Manual reflects NOS 2.0 at PSR level 562. This manual documents the following new features: QAC interface; revised DSP parameter block; new format of GETFNT macro; new macros GETEOJS, SETJOB, SETFS, GETSSID, and GETJOSC; EJT job status and connection status fields; service class types; universal availability of the XJ instruction; binary memory dump request (DMB); new parameter for the STATUS macro that returns additional information for tape files; and revised CIO interface.
B (01-27-83)	Manual updated to reflect NOS 2.1 at PSR level 580 and to make miscellaneous technical changes. New features include extended DSP parameter block, QAC extended GET request, addition to QAC PEEK reply block for remote host files, prefix table for DMB processor, and SHELL and WAIT macros. This edition obsoletes all previous editions.
C (09-30-83)	Manual updated to reflect NOS 2.2 at PSR level 596/587 and to make miscellaneous technical changes. New features include common libraries, enhanced system security, service class assignment by users, expanded equipment status table, and support for CYBER 170 Model 845. This edition obsoletes all previous editions.
D (10-12-84)	Manual updated to reflect NOS 2.3 at PSR level 617 and to make miscellaneous technical changes. New features for this release include alternate CATLIST restriction, default charge restriction, enhanced CLASS command, support of the 834 Disk Subsystems, support of the 639 Magnetic Tape Units, and support of the CYBER 180 Computer Systems. This edition obsoletes all previous editions.
E (03-27-85)	Manual updated to reflect NOS 2.4.1 at PSR level 630 and to make miscellaneous technical changes. New features for this release include support of the CYBER 180 Models 840, 850, and 860, support of the 5870 Printer, and support of the 895 Disk Subsystem.
F (12-16-85)	Manual updated to reflect NOS 2.4.3 at PSR level 647 and to make miscellaneous technical changes. New features include support of CYBER 180 Model 990, support of 836 Disk Subsystems, Control Byte 10 to control CDCNET terminal attributes, and disk error recovery enhancements.
G (09-30-86)	Manual updated to reflect NOS 2.5.1 at PSR level 664 and to make miscellaneous technical changes.

Publication No.
60459690

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

Address comments concerning this manual to:

Control Data Corporation
Technology and Publications Division
4201 North Lexington Avenue
St. Paul, Minnesota 55126-6198

or use Comment Sheet in the back of this manual.

LIST OF EFFECTIVE PAGES

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

PAGE	REV
Front Cover	-
Inside Front Cover	G
Title Page	-
2	G
3	G
4	G
5	F
6	E
7	G
8	F
9	G
10	F
11	G
12	G
13	G
1-1	C
1-2	D
1-3	A
1-4	B
1-5	A
1-6	C
1-7	A
1-8	B
1-9	F
1-10	B
1-11	F
1-12	B
1-13	F
1-14	B
1-15	A
1-16	A
2-1	B
2-2	G
2-3	G
2-4	G
2-5	G
2-6	C
2-7	A
2-8	G
2-9	A
2-10	C
2-11	B
2-12	C
2-13	B
2-14	A
2-15	A
2-16	F
2-17	F
2-18	B
2-19	A
2-20	B
3-1	C
3-2	D
3-3	A
3-4	B
3-5	A
3-6	C
3-7	A
3-8	B
3-9	A
3-10	B
3-11	B
3-12	C
3-13	C
3-14	C
3-15	A
3-16	D
3-17	D
3-18	A
3-19	A
3-20	C
3-21	A
3-22	A
3-23	C
3-24	C
3-25	C
3-26	C
3-27	C
3-28	A
3-29	E
3-30	C
3-31	C
3-32	C
3-33	C
3-34	C
3-35	C
3-36	F
3-37	D
3-38	C
3-39	C
3-40	C
3-41	C
3-42	C
3-43	C
3-44	C
3-45	C
3-46	C
3-47	A
3-48	A
3-49	A
3-50	A
3-51	C
3-52	C
3-53	C
3-54	C
3-55	A
3-56	A
3-57	B
3-58	A
3-59	A
3-60	B
3-61	B
3-62	A
4-1	B
4-2	F
4-3	C
4-4	B
4-5	C
4-6	C
4-7	F
4-8	E
4-9	A
4-10	A
4-11	A
4-12	A
4-13	A
4-14	A
4-15	D
4-16	D
4-17	A
4-18	B
4-19	D
4-20	D
4-21	C
4-22	F
4-23	D
4-24	F
4-25	A
4-26	D
4-27	F
4-28	B
4-29	E
4-30	E
5-1	D
5-2	D
5-3	F
5-4	F
5-5	F
5-6	F
5-6.1/5-6.2	F
5-7	A
5-8	B
5-9	F
5-10	F
5-11	F
5-12	B
5-13	D
5-14	F
5-15	D
5-16	D
5-17	D
5-18	B
5-19	C
5-20	B
5-21	G
5-22	D
5-23	G
5-24	D
5-25	C
5-26	D
5-27	D
5-28	C
5-29	E
5-30	E
5-31	C
5-32	D
5-33	D
5-34	D
5-35	B
5-36	D
5-37	F
5-38	D
5-39	D
5-40	D
5-41	D
5-42	C
5-43	C
5-44	C
5-45	C
6-1	G
6-2	D
6-3	D
6-4	A
6-5	F
6-6	F
6-7	C
6-8	A
6-9	B
6-10	B
6-11	D
6-12	A
6-13	C
6-14	B
6-15	A
6-16	C
6-17	D
6-18	B
6-19	A
6-20	C
6-21	C
6-22	E
6-23	B
6-24	B
6-25	E
6-26	C
6-27	B
6-28	B
6-29	A
6-30	A
6-31	B
6-32	D
6-33	C
6-34	G
6-35	C
6-36	D
6-37	D
6-38	D
6-39	C
6-40	D
6-41	D
6-42	E
7-1	D
7-2	G
7-3	G
7-4	E
7-5	G
7-6	G
7-7	G
7-8	G
7-9	G
7-10	G
8-1	D
8-2	D
8-3	D
8-4	G
8-5	A
8-6	C
8-7	E
8-8	E
8-9	E
8-10	E
8-11	D
8-12	D
8-13	D
8-14	G
8-15	G
8-16	G
8-17	G
8-18	G
8-19	G
8-20	E
8-20.1/8-20.2	G
8-21	G
8-22	B
8-23	D
9-1	A
9-2	B
9-3	D
9-4	B
9-5	B
9-6	A
9-7	D
10-1	B
10-2	A
10-3	A
10-4	B
10-5	A
10-6	A
10-7	C
10-8	C
10-9	D
10-10	B
10-11	A
10-12	D
10-13	B
10-14	D
10-15	F
10-16	F
11-1	D
11-2	D
11-3	C
11-4	C
11-5	D
11-6	D

PAGE	REV
11-7	B
11-8	D
11-9	C
11-10	A
11-11	F
11-12	G
11-13	G
11-14	G
11-15	G
11-16	C
11-17	C
11-18	C
11-19	D
11-20	D
11-21	C
11-22	C
11-23	B
11-24	A
11-25	A
11-26	C
11-27	B
11-28	A
11-29	D
11-30	F
11-31	B
11-32	E
11-33	G
11-34	G
11-35	F
12-1	A
12-2	C
12-3	E
12-4	F
12-4.1/12-4.2	F
12-5	B
12-6	D
12-7	F
12-8	F
12-9	G
12-10	G
12-11	G
12-12	F
12-13	G
12-14	G
12-15	G
12-16	G
12-17	G
12-18	G
12-18.1	G
12-18.2	G
12-19	F
12-20	F
12-21	F
A-1	A
A-2	C
A-3	E
A-4	E
A-5	C
A-6	C
A-7	C
A-8	C
A-9	C
A-10	C
A-11	C
A-12	E
A-13	E
A-14	C
A-15	C
B-1	A
B-2	F
B-3	F

PAGE	REV
B-4	F
B-5	F
B-6	F
B-7	F
B-8	F
B-9	F
B-10	F
B-11	F
B-12	F
B-13	F
B-14	F
B-15	F
B-16	F
B-17	F
B-18	F
B-19	F
B-20	F
B-21	F
B-22	F
B-23	F
B-24	F
B-25	F
B-26	F
B-27	F
B-28	F
B-29	F
B-30	F
C-1	C
C-2	E
C-3	E
C-4	E
C-5	E
C-6	E
C-7	E
C-8	E
C-9	E
C-10	E
C-11	G
C-12	E
C-13	E
C-14	E
C-15	E
D-1	D
E-1	C
E-2	C
E-3	B
E-4	D
E-5	D
E-6	C
E-7	G
E-8	F
E-9	E
E-10	G
E-11	G
E-12	G
F-1	B
F-2	C
F-3	C
F-4	C
F-5	D
F-6	D
F-7	D
F-8	D
F-9	C
F-10	C
F-11	A
G-1	D
G-2	A
G-3	A
G-4	A
G-5	A

PAGE	REV
G-6	A
G-7	A
G-8	A
G-9	D
G-10	A
H-1	A
H-2	A
H-3	A
H-4	B
H-5	B
H-6	A
H-7	A
H-8	A
H-9	A
I-1	A
I-2	A
I-3	A
I-4	D
I-5	A
I-6	C
I-7	A
I-8	C
J-1	D
J-2	A
J-3	A
J-4	A
J-5	A
J-6	C
J-7	A
J-8	A
K-1	D
K-2	D
K-3	G
K-4	G
L-1	G
L-2	F
L-3	G
L-4	G
L-5	G
L-6	G
L-7	G
M-1	D
M-2	D
M-3	D
Index-1	G
Index-2	G
Index-3	G
Index-4	G
Index-5	G
Index-6	G
Index-7	G
Index-8	G
Index-9	G
Index-10	G
Comment Sheet	G
Back Cover	-

PAGE	REV

PAGE	REV

MEMORY WORD FORMATS

The following notations are used in memory word formats:

- Bits of memory words are numbered in decreasing order from left to right, starting with 59 on the left and ending with 0 on the right.
- Bytes (12-bit portions) of memory words are numbered in increasing order from left to right, starting with 0 (byte 0 consists of bits 59 through 48, byte 1 consists of bits 47 through 36, byte 2 consists of bits 35 through 24, byte 3 consists of bits 23 through 12, and byte 4 consists of bits 11 through 0).
- Fields that contain only capital letters indicate portions of memory that contain the display code values for those letters.
- Diagonal lines () indicate a portion of memory that is not used by the system and may contain any value. These portions may, however, be used by future versions of NOS.
- The word 'reserved' or the phrase 'reserved for Control Data' indicates a portion of memory that is either used internally by the system or is reserved for future use. The phrase 'reserved for installations' indicates a portion of memory that each installation may use in whatever manner it chooses.
- A zero indicates a portion of memory that contains all binary zeros.

CHARACTER SETS

The ASCII character set is used in the examples in this manual. Other NOS-compatible character sets are described in appendix A.

SUBMITTING COMMENTS

The last page of this manual is a comment sheet. Please use it to give your opinion on the manual's usability, to suggest specific improvements, and to report any errors. If the comment sheet has already been used, you can mail your comments to:

Control Data Corporation
Technology and Publications Division ARH219
4201 Lexington Avenue North
St. Paul, MN 55126-6198

Additionally, if you have access to SOLVER, an online facility for reporting problems, you can use it to submit comments about the manual. Use NS2 as the product identifier.

RELATED PUBLICATIONS

The following manuals contain information directly related to the material presented in this volume.

You might also want to consult the NOS System Information Manual. It is an online manual that includes brief descriptions of all NOS and NOS product manuals. You can access this manual by logging into NOS and simply entering the command EXPLAIN.

These manuals are available through Control Data sales offices or Control Data Literature Distribution Services, 308 North Dale Street, St. Paul, MN 55103.

Control Data Publication	Publication Number
BML Message Formats	60459940
CDCNET Terminal Interface Usage Manual	60461530
Common Memory Manager Version 1 Reference Manual	60499200
COMPASS Version 3 Reference Manual	60492600
CYBER Loader Version 1 Reference Manual	60429800
CYBER Record Manager Advanced Access Methods Version 2 Reference Manual	60499300
CYBER Record Manager Basic Access Methods Version 1.5 Reference Manual	60495700
Modify Reference Manual	60450100
NOS Version 2 Administration Handbook	60459840
NOS Version 2 Analysis Handbook	60459300
NOS Version 2 Applications Programmer's Instant	60459360
NOS Version 2 Diagnostic Index	60459390
NOS Version 2 Installation Handbook	60459320
NOS Version 2 Network Terminal User's Instant	60459380
NOS Version 2 Operations Handbook	60459310
NOS Version 2 System Overview	60459270
NOS Version 2 Systems Programmer's Instant	60459370

Appendix K contains a comprehensive bibliography of all Control Data publications related to NOS.

DISCLAIMER

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

CONTENTS

1. PROGRAM/SYSTEM COMMUNICATION 1-1

Function Processors 1-1
System Requests 1-2
System Request Processing 1-3
Issuing RA+1 Requests 1-5
NOS Systems Texts 1-6
Macro Usage 1-7
SYSCOM 1-8
Common Deck Usage 1-9

2. FILE ENVIRONMENT TABLE (FET) 2-1

Circular Buffers 2-1
FIRST Address 2-2
LIMIT Address 2-2
IN Address 2-2
OUT Address 2-2
Data Space 2-3
FET Description 2-4
FET Creation Macros 2-14
FILEB 2-14
FILEC 2-14
RFILEB 2-14
RFILEC 2-15
FET Modification Macro 2-17
SETFET Parameter Processing 2-19
SETFET Examples 2-20

3. INPUT/OUTPUT 3-1

Combined Input/Output (CIO) 3-1
CIO Function Processing 3-4
Accessing Files 3-7
Reading Files 3-8
Writing Files 3-8
Random Request Processing 3-10
CIO Processing on Secured Systems 3-12
Reading Files 3-12
Writing Files 3-12
Creating Files 3-12
Status Returned by CIO 3-12
CIO Open and Close Functions 3-12
OPEN 3-13
Mass Storage Operations 3-14
Unlabeled Tape Processing 3-15
Nonstandard Labeled Tape Processing 3-15
Standard ANSI Labeled Tape Processing 3-15
Extended Label Processing 3-17
CLOSE 3-20
CLOSER 3-23
CIO Read Functions 3-24
RPHR (000) 3-25
READ (010) 3-25
READSKP (020) 3-26
READCW (200) 3-27
READLS (210) 3-29
RPHRLS (230) 3-30
READNS (250) 3-31
READN (260) 3-32
READEI (600) 3-33
CIO Write Functions 3-33
WPHR (004) 3-34
WRITE (014) 3-34
WRITER (024) 3-35
WRITEF (034) 3-35
WRITECW (204) 3-36
REWRITE (214) 3-38
REWRITER (224) 3-39
REWRITEF (234) 3-39
OVWRITE (244, 254) 3-40
WRITEN (264) 3-41
File Positioning Functions 3-42
BKSP (040) 3-42
BKSPRU (044) 3-43
REWIND (050) 3-43
UNLOAD (060) 3-44
RETURN (070) 3-45
POSMF (110) 3-46
Standard ANSI Labeled Multifile Set Processing 3-46
Extended Label Multifile Set Processing 3-47

EVICT (114) 3-51
SKIPF (240) 3-52
SKIPFF (240) 3-52
SKIPEI (240) 3-53
SKIPB (640) 3-53
SKIPFB (640) 3-54
Data Transfer Macros 3-54
READC 3-58
WRITEC 3-59
READH 3-59
WRITEH 3-59
READO 3-60
WRITEO 3-60
READS 3-60
WRITES 3-61
READW 3-61
WRITEW 3-61
CYBER Record Manager I/O 3-62

4. LOCAL FILE MANAGER 4-1

Error Processing 4-2
RENAME (000) 4-2
SETFS (003) 4-3
SETFAL (007) 4-3
LOCK (010) 4-4
UNLOCK (011) 4-4
STATUS (012) 4-4
STATUS (013) 4-5
REQUEST (014) 4-6
REQUEST (015) 4-7
ASSIGN (020) 4-9
ENCSF (022) 4-10
PSCSF (023) 4-10
LABEL (024) 4-11
GETFNT (025) 4-18
PRIMARY (031) 4-26
FILINFO (032) 4-26

5. PERMANENT FILE MANAGER 5-1

Alternate User Access Processing 5-4
Error Processing 5-4
PFM Macro Parameter Conventions 5-10
Auxiliary Device Request 5-14
SAVE (001, CCSV) 5-15
GET (002, CCGT) 5-19
PURGE (003, CCPG) 5-20
CATLIST (004, CCCT) 5-21
PERMIT (005, CCPM) 5-27
REPLACE (006, CCRP) 5-29
APPEND (007, CCAP) 5-30
DEFINE (010, CCDF) 5-32
ATTACH (011, CCAT) 5-36
CHANGE (012, CCCG) 5-38
OLD (021, CCOD) 5-42
SETPFAC (022, CCAC) 5-43
SETPFAL (023, CCAL) 5-44

6. CONTROL POINT MANAGER 6-1

SETPR (001) 6-1
MODE (002) 6-2
SETASL (003) 6-3
SETJSL (003) 6-4
SETTL (003) 6-4
EREXIT (004) 6-5
ROLLOUT (006) 6-7
SETSSM (010) 6-8
ONSW (011) 6-9
OFFSW (012) 6-9
GETJN (013) 6-10
GETPR (015) 6-10
GETEM (016) 6-11
GETASL (017) 6-11
GETJSL (017) 6-12
GETTL (017) 6-12
SETLC (022) 6-13
SETRFL (023) 6-14
GETJCR (024) 6-15
SETJCR (025) 6-16
SETSS (026) 6-16
GETJO (027) 6-17
GETJA (030) 6-17
USECPU (031) 6-18
USERNUM (032) 6-18
GETFLC (033, 055) 6-19
PACKNAM (035) 6-20
PACKNAM (036) 6-21
GETSS (037) 6-21
VERSION (044) 6-22
GETLC (045) 6-23
GETGLS (046) 6-23
SETGLS (047) 6-24
MACHID (050) 6-25
GETACT (051) 6-25
SETMFL (052) 6-27
GETPFP (057) 6-27
GETLOF (061) 6-28
SETLOF (062) 6-28
GETJCI (074) 6-29
SETJCI (074) 6-30
PROTECT (075) 6-31
SETJOB (106) 6-31
GETSSID (110) 6-32

GETJOSC (111) 6-33
SHELL (113) 6-33
GETJAL (116) 6-35
SETJAL (117) 6-35
GETUSC (123) 6-36
SETSC (124) 6-38
GETPAGE (127) 6-39
SETPAGE (130) 6-39
GETLVL (133) 6-40
GETRI (134) 6-40
GETCN (136) 6-41
CSTATUS (137) 6-42

7. FILE ROUTING 7-1

DSP Function 7-1
ROUTE 7-8
Error Processing 7-10

8. QUEUE ACCESS INTERFACE 8-1

QAC Requests 8-10
ALTER Request 8-11
GET Request 8-14
PEEK Request 8-16

9. SYSTEM TIME/DATE REQUESTS 9-1

CLOCK 9-2
DATE 9-2
EDATE 9-3
ETIME 9-4
HTIME 9-5
JDATE 9-5
PDATE 9-6
RTIME 9-6
STIME 9-7
TIME 9-7

10. JOB CONTROL 10-1

Translate Control Statement 10-1
CONTROL (004) 10-2
EXCST (005) 10-5
Checkpoint/Restart 10-5
CHECKPT 10-5
Reprieve Processing 10-8
RPVBLK 10-10
REPRIEVE 10-15

11. SYSTEM/LOADER REQUESTS 11-1

System Requests 11-1
ABORT 11-1
DAYFILE 11-1
ENDRUN 11-5
GETLIDA 11-5
GETMC 11-7
GETSSM 11-8
MEMORY 11-9
MESSAGE 11-10
MOVE 11-12
RECALL 11-13
SUBR 11-13
SYSTEM 11-14
WAIT 11-15
User Field Length Dump Requests 11-16
Dump Extended Memory (DEP) 11-16
Dump Extended Memory with Display Code (DED) 11-16
Binary Dump of Central Memory and Extended Memory (DMB) 11-17
Dump Central Memory (DMP) 11-19
Dump Central Memory with Display Code (DMD) 11-19
Queue File Manager 11-19
RERUN (015) 11-20
NORERUN (016) 11-20
Loader Requests 11-20
OVERLAY 11-21
LOADD 11-24
LOADQ 11-26
Memory Allocation for Overlay Loaders 11-28
Interactive Terminal Requests 11-29
DISTC Macro 11-29
CSET Macro 11-31
PROMPT Macro 11-31
SETSLM Macro 11-32
TSTATUS Macro 11-33
EFFECT Macro 11-35

12. PROGRAM WRITING TECHNIQUES 12-1

Writing Programs under NOS 12-1
Parameter Processing 12-1
Writing Interactive Programs 12-4.1
Conversion Problems 12-4.1
Default File Assignments and Special File Treatment 12-4.1
Special Handling 12-5
Other Special Handling 12-5
Program Control of Terminal Activity 12-6

Control Bytes 12-7
0000 - End-of-Line 12-7
0001, 0002 - End-of-Block 12-7
0003 - Auto Input 12-7
0004 - Log Off User 12-8
0005 - Initiate ASCII Input 12-8
0006 - Initiate Transparent Input Mode 12-8
0007 - Initiate Transparent Output 12-9
0010 - Terminal Redefinition for NAM/CDCNET 12-10
0011 - Initiate ASCII Output 12-11
0012 - Reserved 12-11
0013 - End-of-String 12-11
0014 - Internal End-of-Block 12-11
0015 - Executive Auto Input 12-14
0016 - Terminal Redefinition for NAM/CCP 12-14
Mapping Between Terminal Attributes for NAM/CCP and NAM/CDCNET 12-18.1
Control of Program Execution 12-19
Reprieve Processing for User Breaks 1 and 2 12-21

APPENDIXES

A. CHARACTER SETS A-1
Character Set Anomalies A-2
Character Set Tables A-2
Interactive Jobs A-3
Batch Jobs A-3
Jobs Using Line Printers A-3
Jobs Using Magnetic Tape A-12

B. MESSAGES B-1

C. GLOSSARY C-1

D. INTERPRETIVE MODE READING AND WRITING OF EXTENDED MEMORY D-1

E. SPECIAL USER INFORMATION E-1
Job Communication Area E-1
Exchange Package Area E-4
EJT Connection Status Codes E-6
EJT Job Status Codes E-6
EJT Scheduling Data Field E-7
File Types and Origin Codes E-8
Service Class Types E-9
Default Service Classes E-9
Device Types E-10
Error Flags E-11

F. CPU COMMON DECKS F-1
CPCOM F-1
COMCMAC F-5
COMCCMD F-7
Other Common Decks F-8
SYSLIB F-10

G. BINARY FORMATS G-1
PP - CYBER 180, CYBER 170, CYBER 70, or 6000 Computer Systems PP Absolute G-1
OPL - Modify Old Program Library Deck G-2
Modification Table Format G-3
Text Format G-4
OPLD - Program Library Directory G-5
ULIB - User Library Group G-7
TEXT - Unrecognized as Binary G-9
PROC - Procedure G-9
Compressed Compile File G-10

H. EXAMPLES OF RANDOM I/O H-1

I. PROGRAMMING STANDARDS I-1
Documentation Standards I-1
Design Overview I-1
External Interface I-2
Internal Interface I-2
Detailed Code Analysis I-2
Comment Statement Documentation I-2
Table Generation I-7
Further Considerations I-8
Coding Specifications I-8

J. MAGNETIC TAPE FORMATS J-1

Data Formats J-1
I (Internal) Format J-2
S (Stranger) Format J-3
SI (System Internal) Format J-4
L (Long Block Stranger) Format J-5
F (Foreign) Format J-5
End-of-Tape/End-of-Reel Conditions J-6
HDR1 Label Format J-7

K. BIBLIOGRAPHY K-1

L. SHELL PROCESSING L-1

Shell Program Example Using COMPASS L-1
Shell Program Example Using FORTRAN 5 L-5

M. EXTENDED REPRIEVE PROCESSING M-1

INDEX

FIGURES

2-1 Circular Buffer 2-1
2-2 Read Operation 2-3
2-3 Write Operation 2-4
2-4 Standard FET for Mass Storage and Terminal Files 2-4
2-5 Standard FET for Labeled Magnetic Tape Files (CIO and POSMF) 2-5
3-1 Sample Random Access File Format 3-7
3-2 Modified Sample Random Access File Format 3-10
3-3 Data Transfer Buffer Arrangement 3-55
11-1 Absolute Loader Memory Assignment 11-28
E-1 Job Communication Area E-1
E-2 Exchange Package Area E-4
G-1 Modify Library File Format G-2
G-2 Modification Table Format G-3
G-3 Modify Text Format G-4
G-4 Library File Directory Table G-5
G-5 User Library (ULIB) Format G-7
G-6 ULIB Record Format G-8
G-7 ULIB Deck Entry/Externals Format G-9
G-8 Compressed Compile File Format G-10
H-1 COMPASS Program to Create a Random File H-2
H-2 Input File for Program Creating a Random File (Figure H-1) H-4
H-3 Structure of the Random File Created H-5
H-4 COMPASS Program Using READLS Macro to Retrieve a List of Records from a Random File H-6
H-5 COMPASS Program to Replace Certain Records on a Random File H-8
I-1 Source Code for DOCMENT I-4
I-2 External Documentation of Program I-5
I-3 Internal Documentation of Program I-6
L-1 Shell Program Example Using COMPASS L-3
L-2 Shell Program Example Using FORTRAN 5 L-6
M-1 Extended Reprieve Example M-2

TABLES

2-1 Effects of r, w, and rr FET Parameters on File Manipulation Operations 2-13
3-1 CIO Function Status Response Codes and Processing Options 3-6
10-1 RPV Error Codes, Classes, and Flags 10-14
12-1 NAM/CDCNET Terminal Redefinition Parameters 12-12
12-2 NAM/CCP Terminal Redefinition Parameters 12-15
12-3 Terminal Attribute Mapping from NAM/CCP to NAM/CDCNET 12-18.1
A-1 Character Sets for Interactive Jobs A-5
A-2 Character Sets for Batch Jobs A-7
A-3 ASCII to 6/12-Bit Display Code Conversion A-10
A-4 Nine-Track ASCII Coded Tape Conversion A-13
A-5 Nine-Track EBCDIC Coded Tape Conversion A-14
A-6 Seven-Track Coded Tape Conversion A-15

FILE ENVIRONMENT TABLE (FET) 2

This section describes the file communication area necessary for file creation and the processing of many user requests. The file environment table (FET) is the standard communication area or parameter block for the system file processors. The COMPASS programmer must define the FET, whereas the higher level languages (COBOL and FORTRAN, for example) automatically establish and use this area.

Depending on the processor being used, certain areas of the FET must be defined and used in communicating with that processor. The minimum length of a FET is five words, although many processors require a longer FET.

CIRCULAR BUFFERS

A circular buffer is a central memory storage area that contains data during input/output operations. It is called a circular buffer because routines that process input/output treat the first word of the buffer area as if it were contiguous to the last word of the buffer area.

All input/output is performed by passing data between a circular buffer in central memory and a peripheral device (mass storage or magnetic tape, for example).

The buffer pointers FIRST, IN, OUT, and LIMIT define the circular buffer (figure 2-1).

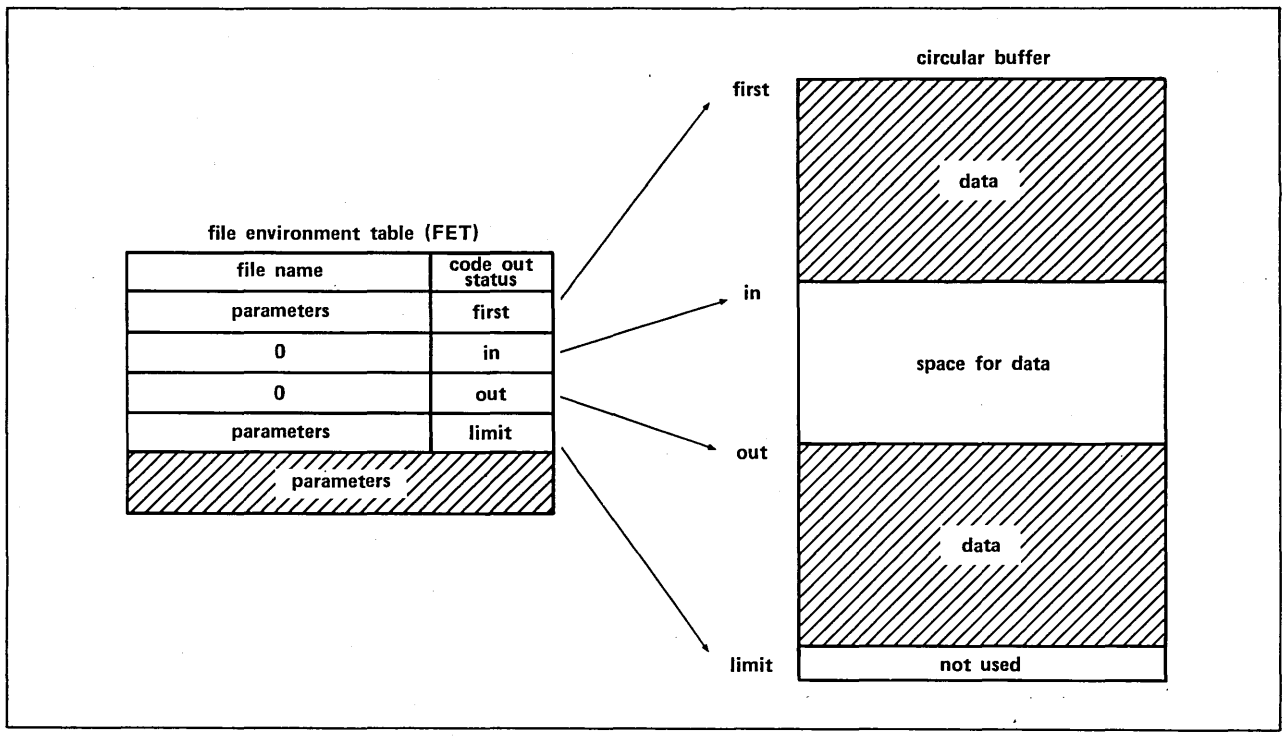


Figure 2-1. Circular Buffer

FIRST ADDRESS

FIRST is the address of the first word of the buffer area. Routines that perform input/output never change the value of FIRST.

LIMIT ADDRESS

LIMIT is the address of the word following the last word of the buffer area. Buffer data is not stored in LIMIT. When LIMIT is reached during a read or write operation, the next word to be read or written is FIRST. Routines that perform input/output never change the value of LIMIT.

The addresses FIRST and LIMIT define the boundaries of the circular buffer. The size of the buffer is always (LIMIT-FIRST).

IN ADDRESS

IN is the address of the next word of the buffer into which data can be entered. During a read operation, the system enters data into the buffer beginning at IN; during a write operation, the user program enters data into the buffer beginning at IN. During a read operation, the system increments IN as it enters data into the buffer; when IN reaches LIMIT, the system immediately resets IN to FIRST. During a write operation, the user program must increment IN as it enters data into the buffer, and must reset IN to FIRST when IN reaches LIMIT (the data transfer macros described in section 3 increment and reset the buffer pointers as necessary for the user program).

If, during a CIO operation, IN reaches (OUT-1), or IN reaches (LIMIT-1) and (OUT=FIRST), the buffer is full. The buffer is full when all words except one contain data. Because the system enters data into the buffer one PRU at a time rather than one word at a time for some CIO operations, the system may consider the buffer to be full if it attempts to enter data into the buffer and the buffer contains less than one PRU of available space.

OUT ADDRESS

OUT is the address of the next word of the buffer from which data can be read. During a write operation, the system removes data from the buffer beginning at OUT; during a read operation, the user program removes data from the buffer beginning at OUT. During a write operation, the system increments OUT as it removes data from the buffer; when OUT reaches LIMIT, the system immediately resets IN to FIRST. During a read operation, the user program must increment OUT as it removes data from the buffer, and must reset OUT to FIRST when OUT reaches LIMIT (the data transfer macros described in section 3 increment and reset the buffer pointers as necessary for the user program).

If OUT reaches IN during a CIO operation, the buffer is empty. Because the system removes data from the buffer one PRU at a time rather than one word at a time for some CIO operations, the system may consider the buffer to be empty if it attempts to remove a PRU of data from the buffer and the buffer contains less than one PRU of data.

Figure 2-2 is an example of a read operation.

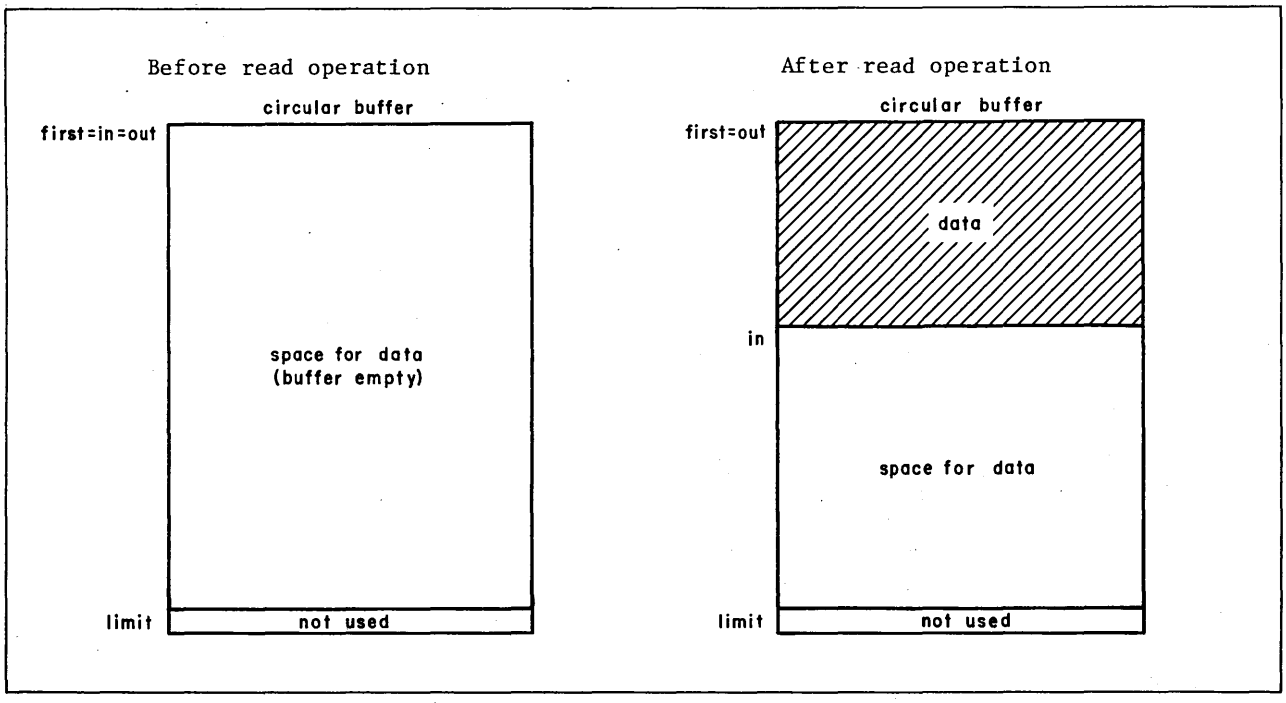


Figure 2-2. Read Operation

DATA SPACE

At any instant a circular buffer is either empty, partially full, or full. When empty, the available unused space in a buffer is one word less than the size of the buffer. When full, all words but one of the buffer contain data.

The system does not permit a circular buffer to be completely filled with data because the IN and OUT pointers would contain equal values and a full buffer could not be distinguished from an empty buffer.

If IN is less than OUT, the amount of data in the buffer is (LIMIT-FIRST) - (OUT-IN) and the available space is (OUT-IN-1). If IN is greater than OUT, the amount of data in the buffer is (IN-OUT) and the available space is (LIMIT-FIRST) - (IN+1-OUT).

Figure 2-3 is an example of a write operation.

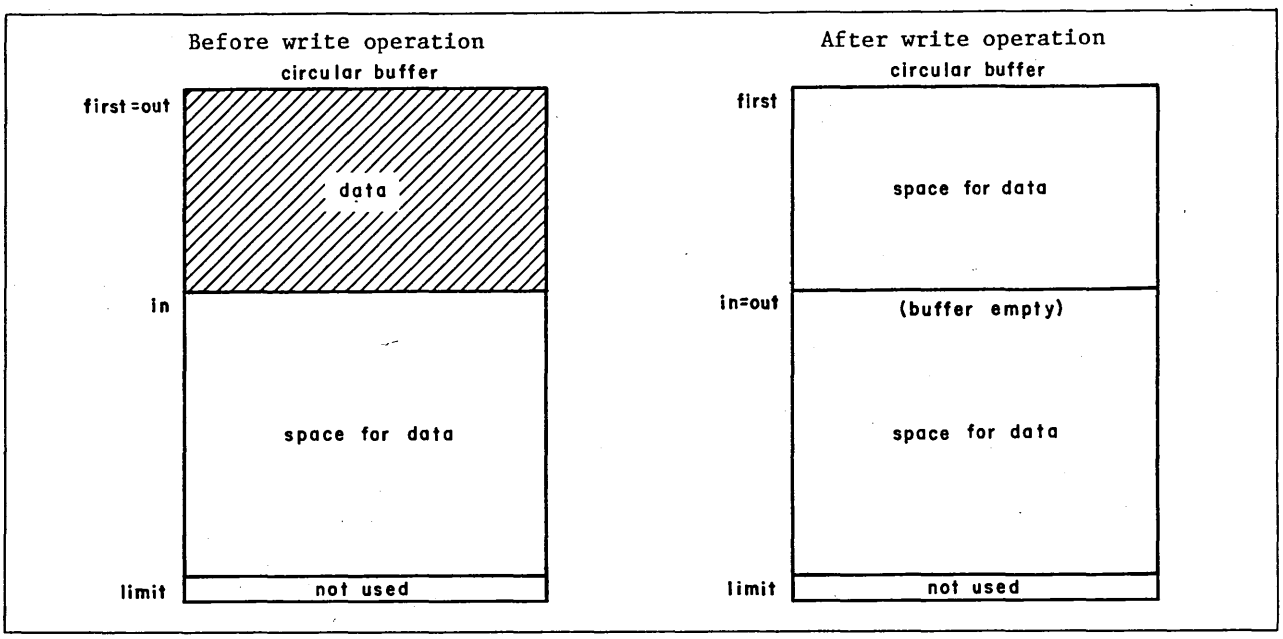


Figure 2-3. Write Operation †

FET DESCRIPTION

There are two basic FET formats. Figure 2-4 illustrates the standard FET for mass storage and terminal files; figure 2-5 illustrates the standard FET for magnetic tape files. The figures are followed by a description of the FET fields. When a field is used by only one of the file processors, it is noted in the description.

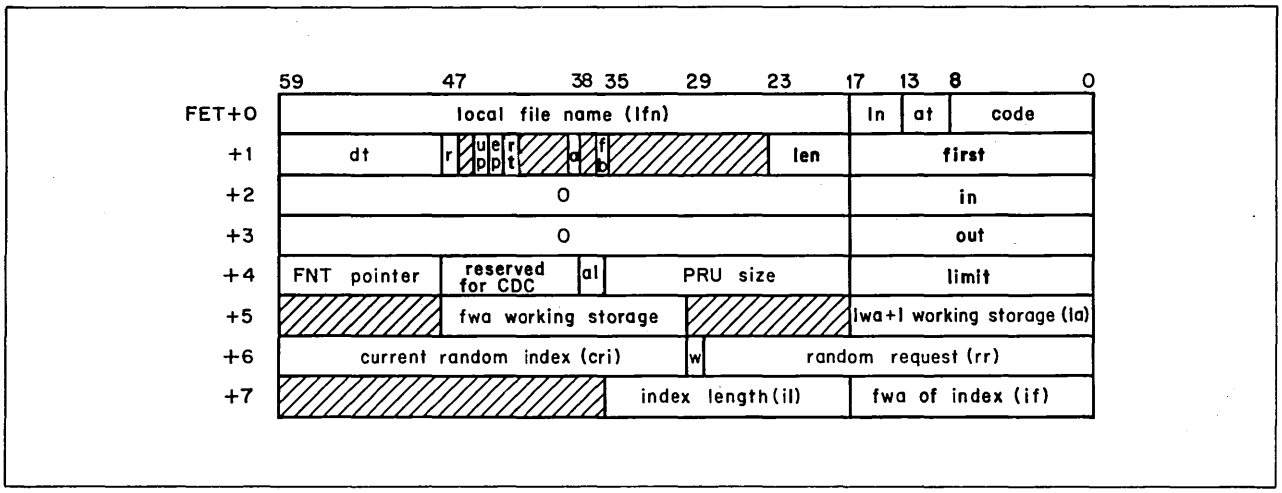


Figure 2-4. Standard FET for Mass Storage and Terminal Files

†Refer to section 5 for a description of the FET fields used in processing permanent file manager (PFM) requests.

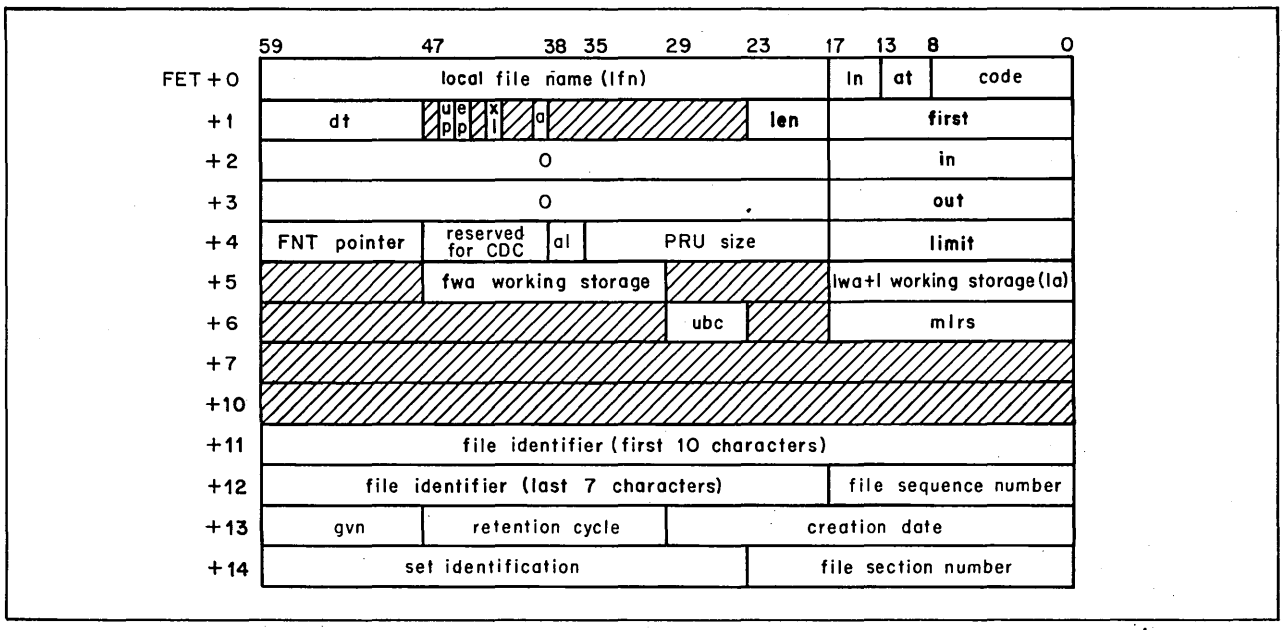


Figure 2-5. Standard FET for Labeled Magnetic Tape Files (CIO and POSMF)†

Parameter	Word (FET+n)	Position	Description
Local file name (lfn)	0	59-18	The lfn field contains one to seven alphanumeric display code characters, left-justified; unused characters are zero-filled. The lfn is the common reference point for all system communication concerning the file.
Level number (ln)	0	17-14	This is the level number for an end-of-record (EOR)/end-of-file (EOF) operation on the file. NOS uses this field for CIO operations and for distinguishing interactive input from noninteractive input (refer to CIO in section 3 and the discussion on writing interactive programs, section 12).
Abnormal termination codes (at)	0	13-9	Status information returned by the function processor when an abnormal situation or error occurs. This field is usually set by the processor when the error processing bit (ep) is set in FET+1. For some processors, at is returned in bits 17 through 9. This field is set to 22_8 by CIO if the ep bit is not set and an error condition is present. Refer to section 3 for the specific error codes returned.

†Refer to the LABEL and OPEN macros for a description of the FET fields used in processing ANSI labels.

Parameter	Word (FET+n)	Position	Description
Code	0	8-0	Request/return code. The user program (or macro) sets this code for the request desired. The function processors alter it only if the request is not completed. For example, the user program requests a read (CIO code 010) but encounters an EOR. CIO returns a status code of 021. These codes are detailed in the function processor descriptions (refer to section 3). The following are subfields for the code parameter. Bit 1: file mode (fm). File mode for input/output operations on S, L, or SI tape formats only. For S or L format tapes, bit 1 is: 0 Coded 1 Binary Binary mode must be selected for SI tape formats or the program is aborted. Bit 0: interlock (ilk). FET interlock bit. Used to indicate system/user access to data associated with the file. The user program sets this bit to 0 (busy or not complete) and the processor sets it to 1 when the request is completed.
Device type (dt)	1	59-48	The 12-bit display code of the type of device on which the file is or will be residing (refer to Device Types in appendix E). After the file is assigned to a device, the STATUS macro or any CIO function enters the device type in this field and sets bit 59 if the device is a nonallocatable nonmass storage device.
Random access (r)	1	47	This bit is set if random processing is to be performed on the file. If this bit is set, the FET must be at least seven words in length.
User processing (up)	1	45	The user program sets this bit to control end-of-reel or end-of-device processing. When CIO encounters an end-of-reel/end-of-device, it returns an abnormal termination code of 2 (bits 13 through 9 of FET+0). For further information about end-of-reel processing, refer to the CIO CLOSER macro description.

Parameter	Word (FET+n)	Position	Description
Error processing (ep)	1	44	This bit is used to indicate to the function processor that the calling program processes errors that occur, such as parity errors or errors in requests to the file managers. The function processor returns the error code to the at field of FET+0. Refer to the section of this manual relating to the specific processor called for descriptions of the error codes returned. If this bit is not set, the function processor either aborts the job or requests operator intervention. If an unrecovered parity or block too large error occurs during a magnetic tape read operation (with the ep bit set) or if a read parity error occurs on a mass storage read operation (with ep bit set), the system performs the following steps. 1. The data in the bad block is stored in the user program's circular buffer. 2. The value of the IN pointer prior to the read is stored in the next word in the circular buffer (pointer to the beginning of the bad data block). 3. The parity error code is set in FET+0. 4. The IN pointer is updated in the FET. This IN pointer value does not include the additional word (pointer to the beginning of the bad data block) stored in the buffer. 5. The FET completion bit is set. The pointer to the bad data is returned on all reads processed by CIO for a mass storage file. If no data is transferred (correct PRU not read), the pointer points to itself, and no update of IN occurs. If tape error processing is inhibited (refer to the LABEL macro, section 4), the preceding steps are not performed regardless of whether or not the ep bit is set.

Parameter	Word (FET+n)	Position	Description
Real time bit (rt)	1	43	When the real time bit is set, the job will not roll out and will not wait for an inaccessible device. The real time bit may be set directly in the FET by a CPU program or it may be set using one of the FET creation macros: FILEB, FILEC, RFILEB, or RFILEC.
Extended label processing (xl)	1	41	Specifies standard (xl=0) or extended (xl=1) tape label processing.
Access level bit (a)	1	39	If this bit is set, the file's security access level is to be taken from or returned to the al field in FET+4.
Flush bit (fb)	1	36	If this bit is set, the file's circular buffer is to be flushed upon abnormal termination or for terminal files when the job is rolled out. Files that are pointed to by the list of files (refer to the SETLOF macro, section 6) and meet the following criteria are flushed with the appropriate write function. • Original error flag (ef) is less than SPET (refer to EREXIT macro, section 6). • No buffer parameter errors; that is, the entire FET must be within the job's field length as follows: LIMIT.LE.FL OUT.LT.LIMIT OUT.GE.FIRST IN.LT.LIMIT IN.GE.FIRST • The code field of the FET is set to one of the valid function codes that enable implicit terminal output (refer to table 3-1). • No CIO error code is present in the FET. • Data is in buffer.

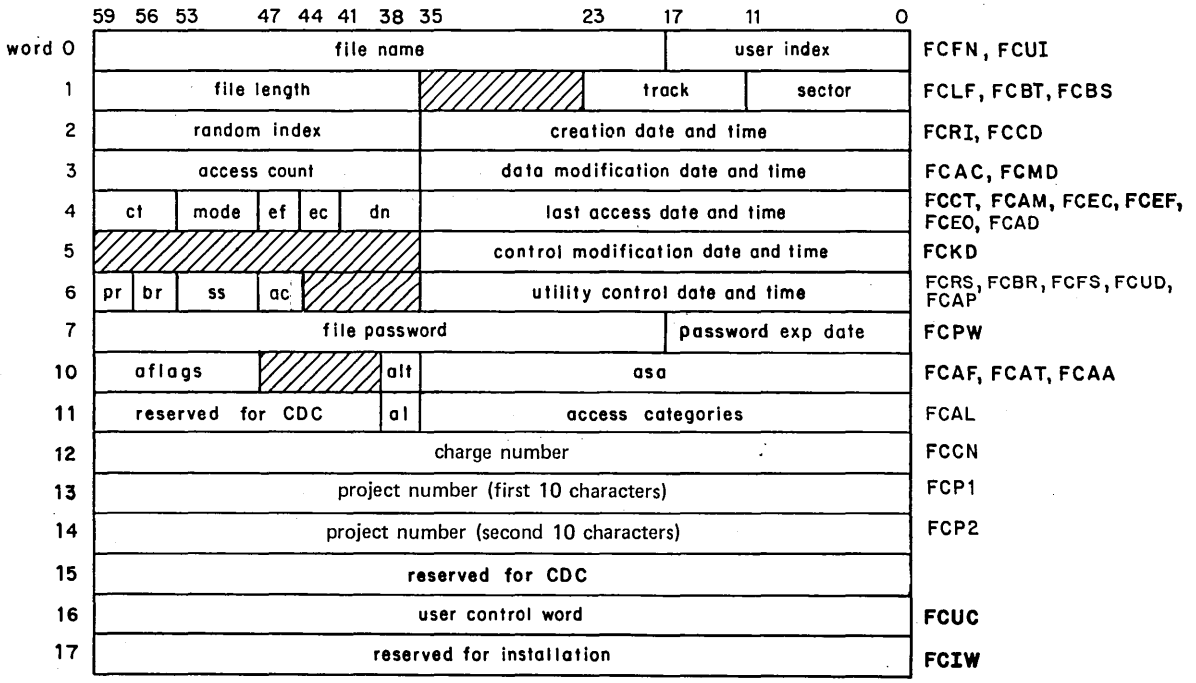
CATLIST (004, CCCT)

The CATLIST macro can be used for the following applications:

- To determine the information about one or more of your own permanent files.
- To determine the information about one or more files belonging to a specified alternate user. You will only be allowed to see information about those files which you can access and which the owner has specified you to receive (via the AC = Y parameter).
- To determine the alternate user information for a specified file that an alternate user can access or has accessed in your catalog (permit data).

The first type of application is the return of the permanent file catalog entries of all your files. If a permanent file name is specified in FET+CFPN, the catalog entry of that one file will be returned.

The format of the permanent file catalog entry is as follows:



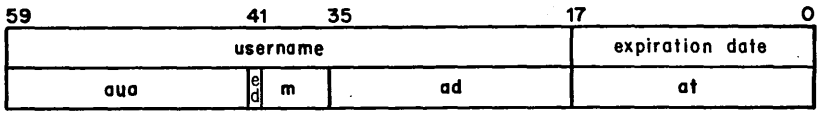
file name	Permanent file name.
user index	User index of file creator.
file length	Length in PRUs of the file.
track	Beginning track of file.
sector	Beginning sector of file (4xxx for a direct access file).
random index	Random disk address of permit sector.

creation date and time	Date and time (yymmddhhmmss in octal) when this file was first entered on the permanent file system. The year (yy) is the calendar year minus 1970.
access count	Count of accesses to file.
data modification date and time	Date and time (yymmddhhmmss in octal) when the data in this file was last modified. The year (yy) is the calendar year minus 1970. For direct access files this field is updated when the file is attached in a modifiable mode, whether or not the file is actually modified.
ct	File category (refer to PFM Macro Parameter Conventions for further information).
mode	File mode (refer to PFM Macro Parameter Conventions for further information).
ef	Error flag: 0 No error. 1 EOI changed by recovery.
ec	Error code: 0 No error. 1 Error in data for file. 2 Error in permit entries for file. 3 Error in data and permit entries for file. 4 File length error. 5-7 Reserved.
dn	Device number (0 through 77_8); each device within a family of permanent file devices is identified by a device number.
last access date and time	Date and time (yymmddhhmmss in octal) when this file was last accessed. The year (yy) is the calendar year minus 1970.
control modification date and time	Date and time (yymmddhhmmss in octal) when the control information (catalog entry and permit record data) for this file was last updated. This corresponds to the last time a CHANGE or PERMIT command or macro was processed for the file. The year (yy) is the calendar year minus 1970.
pr	Preferred residence for file (refer to PFM Macro Parameter Conventions in this section for further information).
br	Backup requirement for file (refer to PFM Macro Parameter Conventions in this section for further information).
ss	Subsystem code for this file (refer to PFM Macro Parameter Conventions in this section for further information).
ac	Alternate CATLIST permission for this file (refer to PFM Macro Parameter Conventions in this section for further information).

utility control date and time	Date and time (yymmddhhmmss in octal) used to determine this file's candidacy for being dumped by permanent file utilities. The year (yy) is the calendar year minus 1970.
file password	Optional password. On a secured system the password is returned as seven asterisks (*******) if a password is defined and the file has an access level higher than that of the job.
password exp date	The date on which the password expires (yymmdd in octal). The year (yy) is the calendar year minus 1970.
aflags	Alternate storage flags; used with the asa field and meaningful only if the file resides on an alternate storage device (as indicated by the alt field); refer to the NOS 2 Analysis Handbook for a complete description of this field.
alt	Alternate storage type: 0 No alternate storage copy of the file exists. 1 An alternate storage copy of the file resides on MSF. 2 An alternate storage copy of the file resides on MSE.
asa	Alternate storage address; meaningful only if an alternate storage copy of the file exists (as indicated by the alt field); if asa is 0, the file does not reside on alternate storage; refer to the NOS 2 Analysis Handbook for a complete description of this field.
al	File's security access level.
access categories	Each of bits 31 through 0, if set, indicates that the corresponding category is in the security access category set for the file. The system OPL common deck COMCVLC can be used to convert an access category value to an access category name.
charge number	Charge number to which the file will be billed.
project number	Project number to which the file will be billed.
user control word	User control information (FET+13_8).

The second type of application is a CATLIST of another user's files. The system returns the permanent file catalog entries of all the user's files you are permitted to access and that are listable by another user. (A file is listable by way of CATLIST only if the owner has specified an alternate CATLIST permission [AC=Y].) If a permanent file name is specified in FET+CFPN, the catalog entry of the specified file will be returned (provided the file is listable by CATLIST and that you are permitted to access it). The following fields are cleared before the catalog entries are returned: user index, access level, access category set, password, and password expiration date.

The third type of application is a CATLIST of PERMIT data. All alternate user access entries (permit entries) for a specified file are returned. The format of an alternate user access entry is illustrated.



username	Alternate user name.
expiration date	The date on which the permit expires [yymmdd in octal; the year (yy) is the calendar year minus 1970].
aua	Number of accesses the alternate user has made to the file (hhmmss in octal).
ed	Bit 41; if set, indicates that an expiration date is present in bits 17-0 of the first word of the entry.
m	Permission mode (bit 40 set if this was an accounting permit and was not created by a PERMIT command or macro; bit 40 clear indicates an explicit permit set by PERMIT command or macro; bits 39 through 36 are same as mode described in first type of request).
ad	The last date the alternate user accessed the file [yymmdd in octal; the year (yy) is the calendar year minus 1970].
at	The time of day the alternate user last accessed the file (hhmmss in octal).

Macro format:†

Location	Operation	Variable
	CATLIST	addr,pfn,un,m,pn,r,fo,dn

addr	Address of the FET for the CATLIST function.
pfn	Address containing the file name; if this parameter is omitted, CATLIST information for all files in your permanent file catalog (or in the catalog specified by un) is returned.
un	Address containing the name of the alternate user catalog to be searched for the catalog information. If un is omitted, your own catalog is searched.
m	If this parameter is the letter M, alternate user access information for file pfn is returned. If this parameter is specified, the pfn is required and un is ignored.

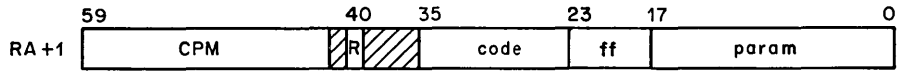
†This macro is not available in SYSTEXT. The user program must specify systems text PSSTEXT or call common deck COMCMAC (refer to appendix F).

CONTROL POINT MANAGER 6

Control point manager (CPM) enables your program to alter or interrogate parameters in your job's control point area.

Unless otherwise noted, all errors encountered by CPM cause the job step to be aborted; no user error processing is available.

The format of the call to CPM is:



R	Auto recall bit.
code	CPM function code.
ff	Subfunction code for functions 003 and 017.
param	Parameter for the function.

Common decks required for absolute assemblies by the functions processed by CPM are COMCCPM and COMCSYS. For relocatable assemblies, these decks are satisfied by default from the library SYSLIB.

SETPR (001)

The SETPR macro decreases the CPU priority of a job. A user job cannot increase its priority.

Upon job initiation, a job is assigned the maximum priority allowed for its service class. (The installation defines these priority values.) If a job's CPU priority is lower than that of other jobs, the job is assigned control of the CPU only when jobs of a higher priority do not need it.

Macro format:†

Location	Operation	Variable
	SETPR	n

n CPU priority; ($1 \leq n \leq 70_8$).

†This macro is not available in SYSTEXT. The user program must specify systems text PSSTEXT or call common deck COMCCMD or COMCMAC (refer to appendix F).

MODE (002)

The MODE macro defines the error conditions that cause the system to exit from normal processing and, on CYBER 180-class models, changes the instruction stack purging status of the central processor (refer to the exchange package description in appendix E).

When the error specified in a MODE macro occurs, the system sets the appropriate error flag and exits from normal processing to perform any error processing required. If an error occurs for which the user program did not select the exit mode processing, the system ignores the error and continues normal processing.

Macro format:†

Location	Operation	Variable
	MODE	m,n,p

m — CPU program error exit mode.

n — This parameter is included for compatibility with earlier versions of NOS. The value supplied is ignored.

p — Instruction stack purging (CYBER 180-class models only):

- 0 Normal instruction stack purging.
- 1 Extended instruction stack purging.

The processor will exit when this condition occurs	If m has one of these octal values
Address out of range (automatically selected for model 176)	1, 3, 5, 7, 11, 13, 15, 17††
Operand out of range (overflow only on model 176)	2, 3, 6, 7, 12, 13, 16, 17
Indefinite operand	4, 5, 6, 7, 14, 15, 16, 17
Underflow (model 176 only)	10, 11, 12, 13, 14, 15, 16, 17

The maximum value and default on all models except model 176 is 7, indicating that the processor exits on all conditions. The maximum value on model 176 is 17_8, indicating that the processor exits on all conditions. The default value on model 176 is 7, indicating that the processor exits on all conditions except underflow.

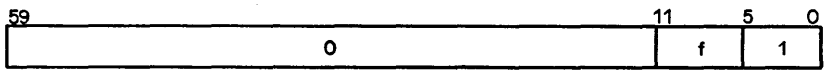
†This macro is not available in SYSTEXT. The user program must specify systems text PSSTEXT or call common deck COMCCMD or COMCMAC (refer to appendix F).

††On a model 176, the address out of range error will always cause a processor exit. Therefore, any attempt to disable this type of exit (by specifying a value not included in this set) on a model 176 will cause an informative message to be issued to the dayfile.

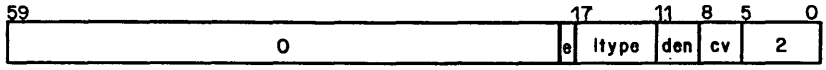
The following keys return the corresponding key words. Except where otherwise indicated in the descriptions, the values returned are those supplied in the ASSIGN, LABEL, or REQUEST function, and may not equal the values actually assigned to the tape file.

Key† — Returned Key Word for Tape File

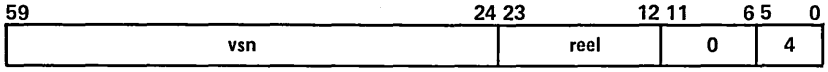
1



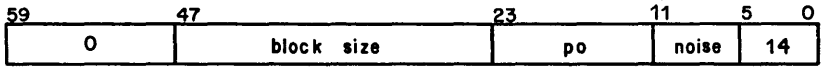
2



4



14_8



Field	Key	Bit(s)	Description
f	1	11-6	Tape format: 0 Internal (I). 1 System internal (SI). 2 Foreign (F). 3 Stranger (S). 4 Long block stranger (L).
e	2	18	If set, error processing is inhibited (same as bit 14 in po field).
ltype	2	17-12	Label type: 0 Unlabeled. 1 Standard label (ANSI 1969 standard). 14_8 Nonstandard label.
den	2	11-9	Tape density: 1 556 cpi (seven-track). 2 200 cpi (seven-track). 3 800 cpi (seven-track) or 800 cpi (nine-track). 4 1600 cpi (nine-track). 5 6250 cpi (nine-track).

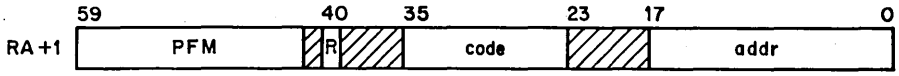
†The symbols for each key value are defined in common deck COMSLFM.

Field	Key	Bit(s)	Description
cv	2	8-6	Conversion mode:† 0 BCD conversion (seven-track). 1 ASCII conversion (nine-track). 2 EBCDIC conversion (nine-track).
vsn	4	59-24	Volume serial number of the currently mounted magnetic tape. This is the actual VSN, not the value from the VSN or LABEL command.
reel	4	23-12	Logical reel number of the currently mounted magnetic tape. Corresponds to the ANSI standard file section number.
block size	14_8	47-24	Maximum block size in central memory words. Always 1000 octal for I and SI format tapes. For F format tapes, the block size is calculated from the FC or C parameter value specified on the tape assignment command (refer to Volume 3, System Commands). For S and L format tapes, the block size returned is the size used on the last CIO operation, as determined from the default block size, mlrs field in the FET, or CIO buffer size (refer to appendix J). The field is zero if no previous CIO function has been issued on the tape file. Refer to appendix J for further information.
po	14_8	23-12	Processing options. Refer to the description of the po field for the LABEL macro in section 4. The 12 bits in each field have the same respective definitions. If po bit 18 (corresponding to po bit 42 for the LABEL macro) is set upon return from the FILINFO macro, GCR hardware write error correction is disabled. If po bit 19 (corresponding to bit 43 for the LABEL macro) is set upon return from the FILINFO macro, the issuing of all error recovery messages to the user dayfile is enabled.
noise	14_8	11-6	Noise size in frames.

If the specified file is not local to your job, words 1 through 4 of the parameter block are cleared. All other fields (including words containing keys) are unchanged.

†Conversion mode of labels (if any) and coded data (if any). This does not indicate whether the data on the tape is coded or binary.

Permanent file manager (PFM) processes all permanent file requests. The format of the RA+1 call to PFM is as follows:

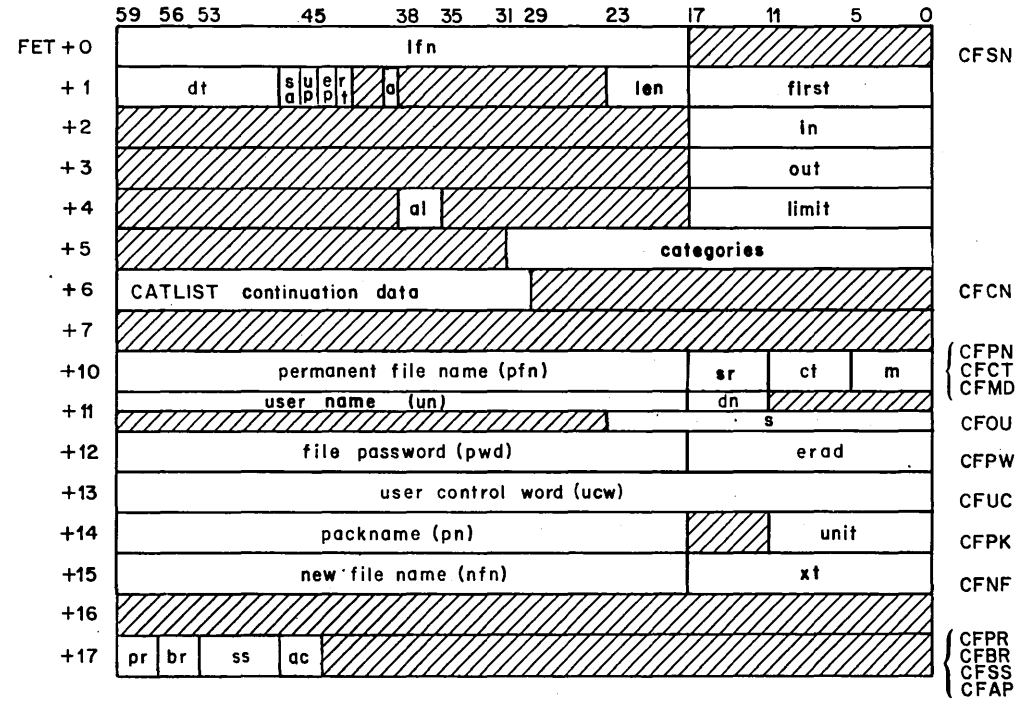


R Auto recall bit.

code Function code.

addr Address of the FET.

The FET used by all PFM requests is formatted as follows. The mnemonics (CFSN, for example) are defined in common deck COMSPFM. The length of the FET is defined by the symbol CFLM.



lfn Local file name (left-justified, zero fill).

dt Device type.

sa Suppress address on error messages (bit 46).

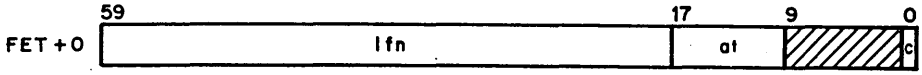
up User processing bit (bit 45).

ep Error processing bit (bit 44).

rt	Real-time processing bit (bit 43).
a	Access level processing bit (bit 39).
len	FET length minus 5.
first	FIRST parameter for file's circular buffer.
in	IN parameter for file's circular buffer.
out	OUT parameter for file's circular buffer.
al	Security access level.
limit	LIMIT parameter for file's circular buffer.
categories	Security access category set (refer to SETPFAC macro).
pfn	Permanent file name (left-justified, zero fill); if pfn is zero, lfn is used as the permanent file name.
sr	Special request subfunction.
ct	File category (refer to Permission Modes, File Categories).
m	File access mode (refer to Permission Modes, File Categories).
un	Alternate user name.
dn	Device number for CATLIST option (range is 1 to 77_8).
s	Number of PRUs (octal) desired for the file.
pwd	Optional file password (left-justified, zero fill).
erad	Error message return address.
ucw	User control word.
pn	Pack name of auxiliary device.
unit	Number of units of multiunit device.
nfn	New file name (refer to CHANGE macro).
xt	Expiration term or date for password or permit.
pr	Preferred residence for file (bits 59-57).
br	Backup requirement for file (bits 56-54).
ss	Interactive subsystem asociated with file (bits 53-48).
ac	Alternate user CATLIST permission (bits 47-46).

For a more detailed description of the FET, refer to the discussion of the FET in section 2.

After a request to PFM is complete, the first word of the FET contains the following information.



lfn Local file name.

at Abnormal termination code.

c Bit 0 is set to 1 upon completion of the request.

The FET length may be five words, if no special options are required, or up to 20_8 words, depending on the special options required.

When a PFM macro request is processed, the parameter values specified are placed in their corresponding fields in the FET.

On a macro call, the register X1 may be used to specify the permanent file name (pfn); the register X2 may be used to specify the file password (pwd); and the register X3 may be used to specify the alternate user name (un). After the macro is processed, the register X2 contains the address of the FET.

The address of the FET must be supplied in the addr parameter. The call to PFM must be made with the auto recall bit set (the PFM macros automatically do this).

The system OPL common decks required for an absolute assembly of a program containing PFM requests are COMCPFM and COMCSYS. In order to use the mnemonics defined for PFM FET addresses, PFM addresses, permission modes, file categories, or error codes, the system OPL common deck COMSPFM is required for absolute or relocatable assemblies. Function mnemonics are specified in the macro descriptions. If the al parameter is specified on a SAVE, DEFINE, or SETPFAL macro, the system OPL common deck COMSMLS is required.

On a secured system, your job may not access a permanent file unless it is validated for the access level and access category set of the file. For an alternate-user access, such an invalid access attempt is treated as a file-not-found error; for an access to one of your own files, it is treated as a security conflict (refer to section 1 for a description of security conflict processing).

ALTERNATE USER ACCESS PROCESSING

Another user can access one of your permanent files only if you explicitly allow it. There are two ways you can allow alternate user access:

- By declaring the file to be a public or semiprivate file. These types of files may be accessed by all users in your family; the users may only access the file in the mode you specify (and in those modes that are subsets of the specified mode; refer to PFM Macro Parameter Conventions in this section for a description of the hierarchy of file access modes).

- By permitting a specific user to access the file with a particular access mode (by using the PERMIT command or macro). This may be done on a private or semiprivate file; on a semiprivate file, the access mode specified by PERMIT overrides the default semiprivate access mode for the file. An expiration date may be assigned to each permit entry; when this date is reached, the specified user is no longer able to access the file.

In addition, you may assign a password to each of your files. An alternate user must specify this password in order to access the file. This password may also have an expiration date; when this date is reached, alternate users are no longer able to access the file.

ERROR PROCESSING

The user program can issue requests to PFM and have control returned if certain error conditions occur. To do this, the error processing bit (ep) must be specified in FET+1. The error codes are returned in the abnormal termination (at) code field of FET+0 (bits 17 through 10). If the user program specifies erad in FET+12_8, the error message is returned at this address instead of being issued to the user's dayfile. A maximum of three central memory words are returned. System errors are issued to the system and error log dayfiles regardless of the erad specification.

The following error codes are returned in the abnormal termination code field. It is recommended that the error code mnemonics defined in common deck COMSPFM be used in testing these error conditions. The mnemonics are qualified by symbol ERRMSG. For example, to set the B7 register to the value of the FBS mnemonic, use the following instruction.

```
SB7 /ERRMSG/FBS
```

Error Code†	Mnemonic	Description
1	FBS	The specified direct access file is attached with conflicting mode (file busy).
2	FNF	One of the following: • The specified permanent file could not be found. • The specified user name could not be found. • You are not authorized to access the specified file. • The specified local file could not be found (SAVE, REPLACE, or APPEND). • An indirect access file request was issued for a direct access file. • A direct access file request was issued for an indirect access file.
3	EFL	The file specified on a SAVE, REPLACE, or APPEND macro contains no data.
4	NMS	The file to be saved is not on mass storage; the first track of the file is not recognizable.
5	FAP	The user program has already saved or defined a file with the name specified.

†Codes are given in octal.

Error Code†	Mnemonic	Description
6	IFT	The user program attempted to define a file that is not a local file.
7	FNE	File name contains illegal characters.
10	IUA	You are not validated to create direct access or indirect access files or to access auxiliary devices.
11	IDR	The device type (r parameter in macro calls) specified on a request for an auxiliary device cannot be recognized or does not exist in the system. If the removable auxiliary device specified by the pn parameter is not the same type as the system default, the r parameter must be included; if not, this error code is returned.
12	FTL	The local file specified for a SAVE, REPLACE, or APPEND request exceeds the length allowed, or the direct access file specified for an ATTACH in write, modify, update, or append mode exceeds the direct access file length limit for which the user is validated.
13	ILR	One of the following: • Illegal function code passed to PFM. • Illegal permit mode or catalog type specified. • CATLIST request has permit specified without a file name. • PERMIT attempted on a public file. • DEFINE, CHANGE, or SAVE attempted with alternate user name in FET.
14	PFN	Access to the permanent file device requested is not possible.
15	DAD	The device on which the file resides may not contain direct access files because: • The device is not specified as a direct access device in the catalog descriptor table. • The device is not specified as ON and initialized in the catalog descriptor table. • The device is a dedicated indirect access permanent file device.

†Codes are given in octal.

Error Code†	Mnemonic	Description
16	PFA	Because a permanent file utility is currently active, the operation is not attempted; the user program should retry the operation.
17	DTE	An error occurred in a read operation during a file transfer.
20	COF	The number of files in your catalog exceeds the limit (refer to the LIMITS command in Volume 3, System Commands).
21	COS	The cumulative size of the indirect access files in your catalog exceeds the limit (refer to the LIMITS command in Volume 3, System Commands).
22	SPN	The number of PRUs specified via the s parameter on the DEFINE macro is not available.
23	IOE	A request is attempted on a local file that is currently active. This error can occur, for example, if the user program creates two FETs for the same file and issues a second request before the first is complete.
24	--	The job's local file limit has been exceeded by an attempt to GET or ATTACH the file.
25	PRL	The job's mass storage PRU limit is exceeded during preparation of a local copy of an indirect access file.
26	PLE	Permit limit is exceeded for a private file.
27	PAE	One of the following: • An illegal function code was specified. • The FET address specified was not within the user program's field length. • The error message return address specified was not within the user program's field length. • PFM was not called with auto recall specified.
30	RSE	The resource executive detected a fatal error.
31	TKL	No available space remains on the device.

†Codes are given in octal.

Error Code †	Mnemonic	Description
32	FLE	The length of a file does not equal the catalog length; the action taken depends on the type of request issued. Request — Action GET — A local file is created with length being the actual length retrieved. SAVE — If file length is longer than TRT specification, file is truncated. REPLACE — Same as for SAVE.
33	RIN	PERMIT random address error.
34	DAF	The system sector data for the file does not match the catalog data.
35	RPE	The same file is found twice during a catalog search. This error can occur for APPEND or REPLACE requests after a file is found and purged and the catalog search is continued.
36	ABT	Error flag detected at PFM control point.
37	MSE	An error is encountered in reading or writing a portion of the permanent file catalog or permit information. A write error encountered during the device to device tansfer of an indirect access file.
40	EDA	Error in data for file.
41	EPT	Error in permit entries for file.
42	EDP	Combination of errors 40 and 41.
43	FLC	An error in the EOI position of the file was detected by recovery routines.
44	NEM	PFM detected a fatal RESEX error. PFM does not issue a message since RESEX has issued an appropriate message. If the error processing (ep) bit is set and the erad field of FET+12$_8$ contains a valid address, the message RESEX issues is returned to that address.
45-47	—	Reserved.
50	FSE	Verification of the file structure or owner identification failed.

† Codes are given in octal.

Error Code†	Mnemonic	Description
51	SSE	Error in system sector of indirect access file.
52	BCS	An invalid catalog or permit sector was detected while attempting to access the file.
53-70	—	Reserved.
71	PPE	The file does not reside on disk. An unrecoverable error was detected when attempting to stage the file to disk from alternate storage.
72	PSI	The file does not reside on disk. It is currently being staged from alternate storage to disk.
73	PWE	The file must be staged from alternate storage to disk, but the subsystem is not currently available to perform the stage operation.
74	FIN	The file is currently not available for write, modify, update, or append modes of access because a system utility has the file attached in utility mode.
75	—	Reserved.
76	—	Reserved.
77	FEO	The local file specified for a SAVE, REPLACE, or APPEND request was an execute-only file.
100	SGD	The file does not reside on disk and must be staged from alternate storage to disk.
101	IPA	The device number, track, and sector specified as the permanent file catalog (PFC) address are invalid.
102	PVE	The creation date and time, user index, or alternate storage address do not agree with the current PFC contents.
103	FND	Access to a file that does not reside on disk is not permitted with the current function.
104	INA	A software interlock is currently not available to a request specifying both ep and up processing options.
105	AIO	The disk space for a file cannot be released when the alternate storage image is labeled obsolete or the alternate storage address is not specified in the PFC.

† Codes are given in octal.

Error Code†	Mnemonic	Description
106	ASE	The disk space for a file cannot be released when a permanent error status is set for the alternate storage file image.
107	FTF	File name table (FNT) space is currently not available for a request that has both ep and up processing options set.
110	ICU	Verification of the PFC prohibits the setting of a new alternate storage address when the alternate storage address is defined and not labeled obsolete. Also, an existing valid disk address cannot be replaced in the PFC entry.
111	PEA	The PFM activity count is currently at its limit. This condition is reported to the requesting program only if both the ep and up processing options are set.
112	NVX	You are not authorized to set a password or permit expiration date.
113	DEM	The expiration date or term you specified exceeds the maximum expiration term allowed by your site.
114	JCA	Your job is not validated for the access level or access category set for the specified file.
115	LNJ	The specified access level is not valid for the calling job.
116	WDP	The local file has a higher access level than the permanent file it is replacing or is being appended to, or the local file has a lower access level than the job on a DEFINE request. You do not have the special authorization to perform this operation.
117	CNJ	The specified access category set is not valid for the calling job.
120	LND	The local file access level is incompatible with the user's master device access level limits.
121	NVD	The specified access level is lower than the current file access level, and you are not authorized to lower file access levels.
122	NTD	No temporary device could be found with the correct access level.
123	LNF	The specified access level on a SAVE request is lower than the local file access level.
124	TPE	The system encountered a temporary error that prevents it from moving your file in from alternate storage. Wait and try the request again.

†Codes are given in octal.

PFM MACRO PARAMETER CONVENTIONS

Several methods are available to you for specifying the m, ct, ss, br, pr, and ac parameters in PFM macros. You can either specify an alphabetical key or an address that contains an integer value that corresponds to a key. If an address is specified, the value must be right-justified and zero-filled in a 60-bit word.† The values can be established with the mnemonics in the following list. The user program must call common deck COMSPFM to use these mnemonics (relocatable or absolute assembly). The valid mnemonics (if any), keys, and values for the m, ct, ss, br, pr, and ac macro parameters are listed.

Parameter	Mnemonic	Key	Value (Octal)	Description
m				File or user permission mode:
	PTWR	W	0	Allows the user program to write, read, append, execute, modify, and/or purge the file. New information can be added within the existing boundaries of the file, and the file can be extended or shortened. This mode can be specified for direct or indirect access files.
	PTRD	R	1	Allows the user program to read and/or execute the file. This mode can be specified for direct or indirect access files.
	PTAP	A	2	Allows the user program to append information to the end of the file (EOI). This mode can be specified for direct or indirect access files.
	PTEX	E	3	Allows the user program to execute the file. This mode can be specified for direct or indirect access files.
	PTNU	N	4	Removes permission previously granted via PERMIT macros. This mode can be specified for direct or indirect access files.

†These parameters are stored in the FET in right-justified, zero-fill format, except for the lfn, pfn, pwd, and nfn parameters which are stored in left-justified, zero-fill format.

Parameter	Mnemonic	Key	Value (Octal)	Description
	PTMD	M	5	Allows the user program to modify, append, read, and/or execute a direct access file. Adding new information within the existing boundaries of the file is legal, and the file size may be extended, but the file may not be shortened. For an indirect access file, this mode allows you to perform all the functions of the R (read) mode and the A (append) mode.
	PTRM	RM†	6	Allows the user program to read and/or execute a direct access file with the implication that another user may currently be accessing the same file in M (modify), A (append), or U (update) mode. For indirect access files, this mode is the same as R (read) mode.
	PTRA	RA	7	Allows the user program to read and/or execute a direct access file with the implication that another user may currently be accessing the same file in A (append) mode. For indirect access files, this mode is the same as R (read) mode.
	PTUP††	U	10	Allows the user program to modify, read, and/or execute a direct access file. Information may be added within the existing boundaries of the file, but the file size must be maintained. For indirect access files, this mode is the same as R (read) mode.
	PTRU	RU	11	Allows the user program to read and/or execute a direct access file with the implication that another user may currently be accessing the same file in U (update) mode. For indirect access files, this mode is the same as R (read) mode.

†Special care should be taken when using modes RM, RA, or RU. Programs using access techniques (either CIO or CYBER Record Manager) that do not expect concurrent updating of a file may get erroneous results if these modes are used. CRM Advanced Access Methods (refer to the CYBER Record Manager Advanced Access Methods Reference Manual) do not expect concurrent updating of a file by another user. Therefore, if a file has been attached in RM, RA, or RU mode and you are using these access methods, a warning message is issued to you stating that the file is bad when, in fact, it is not.

††The CIO function WRITECW cannot be used to perform a write on a file attached in U (update) mode.

Parameter	Mnemonic	Key	Value (Octal)	Description
ct				File category:
	FCPR	P	0	Private. Private files are available for access only by the originator or those to whom the originator has explicitly granted permission (refer to the PERMIT macro).
	FCSP	S	1	Semiprivate. Semiprivate files are available for access by all users who know the file name, user name, and file password. The file owner specifies the default permitted mode of access for all users. The file owner may also specify, via the PERMIT macro or command, different permissions for different users. The system records in the file owner's catalog the user name of each user who accessed the file, the number of accesses, and the date and time of the last access.
	FCPB	PU	2	Public. Public files are available for access by all users who know the file name, user name, and file password. The file owner specifies the permitted mode of access for all users. The system records the number of times the file was accessed but does not record user names or the last access date and time.

Parameter	Mnemonic	Key	Value (Octal)	Description
ss				Subsystem indicator:
	--	NULL	0	Null subsystem.
	--	BASIC	1	BASIC subsystem.
	--	FORT	2	FORTRAN 5 subsystem.
	--	FTNTS	3	FORTRAN Extended 4 subsystem.
	--	EXEC	4	Execute subsystem.
	--	BATCH	5	Batch subsystem.
br†				Backup requirement:
	BRNO	N	1	Backup is not required on a dump tape.
	BRAL	Y	2	Backup is required on a dump tape.
	BRMD	MD	3	Backup is required on a dump tape only if the file is not resident on alternate storage.
pr†				Preferred residence:
	RSLK	L	1	The file will be locked to disk. (Special validation is required for this option.)
	RSDS	D	2	The file should reside on disk at all times.
	RSMS	M	3	The file should reside on alternate storage when it is not being accessed.
	RSNP	N	4	No preferred residence for the file.
ac				Alternate CATLIST permission:
	ACNO	N	1	Alternate users may not obtain information about the file by using CATLIST (default value).
	ACYS	Y	2	Alternate users may obtain information about the file by using CATLIST.

†If this parameter is omitted or a value of 0 is specified, PFM uses the installation default value.

AUXILIARY DEVICE REQUEST

Unless you explicitly declare otherwise, all of your permanent files reside on family devices. As stated in Volume 3, System Commands, you may wish to supplement the mass storage provided by your family devices by retaining your files on auxiliary devices. (Not all users may be validated to use auxiliary devices.) There are four parameters (pn, r, un, and pwd) that uniquely identify file pfn on an auxiliary device.

- The pn parameter specifies the one- to seven-character system-defined pack name of the auxiliary device. The device can be either public or private, as defined by the installation.

- The r parameter specifies the type of auxiliary device on which the file resides or is to reside. An auxiliary device is any supported device that an installation defines as an auxiliary; it need not be a removable device.

  r may be DBn, DDn, DIn, DJn, DKn, DLn, DMn, DQn, DV, or DW, as described for Equipment Codes in appendix E.

  If you need two or more removable auxiliary devices, or one removable auxiliary device and one or more magnetic tapes, at any one time during your job, a RESOURC command must be included (refer to Volume 3, System Commands).

  An installation can provide additional continuous storage on a DB, DD, DI, DJ, DK, DL, DM, or DQ type device by combining several physical units into one logical unit. A device so defined is known as a multiunit device. To specify such a device, the r parameter must include the number of units. For example, if four 844-21 units have been combined as one multiunit device, the r parameter must be DI4. If it is not, the job step aborts.

  However, if r is DB, DD, DI, DJ, DK, DL, DM, or DQ but n is omitted, the unit count is assumed to be 1.

  The r parameter is required only if the desired device has a device type different from that of the installation-defined default device type, and the installation has defined the desired device as removable. However, you can be assured of always accessing the proper device by specifying this parameter. If an incorrect value is specified for r, PFM error message 11_8 (IDR) is issued. For all PFM macros, if pn is specified but the device is not available, the job is aborted. By setting the error processing bit (FET+1, bit 44) you can bypass the abort and direct the system to make the specified device available. If both the error processing bit and the user processing bit (FET+1, bit 45) are set, control is returned to the user program when the device is unavailable and error code 14_8 (PFN) is set in FET+0.

- The un parameter specifies the one- to seven-character optional user name. If the device is public, files are accessed in the same manner as specified for family devices; that is, the un parameter must be included only if you wish to access files that another user has explicitly or implicitly permitted you to use. If the device is private, there is only one owner. All other users who have the proper validation can access files on the device, but the system prevents them from creating files on that device.

- The pwd parameter specifies the one- to seven-character password associated with the file. As with files on family devices, the pwd parameter must be included only if the originator requires that a password be specified.

SAVE (001, CCSV)

The SAVE macro enables the user program to retain a copy of a temporary mass storage file as an indirect access permanent file. The original file is rewound when the copy is complete.

Macro format:

Location	Operation	Variable
	SAVE	addr,pfn,pwd,ucw,ct,m,pn,r,fo,ss,br,pr,xt,al,ac

addr — Address of the FET; the local file name must be set in FET+0.

pfn — Address containing the name of the file; name the file is given in the permanent file catalog.

pwd — Address containing the password to be placed with the file in the permanent file catalog.

ucw — Address containing user control word to be placed with the file in the permanent file catalog. If this word exists in the catalog, it is returned to FET+13_8 when the file is accessed (default=0).

ct — File category:

ct	Description
P	Private file (default value).
S	Semiprivate.
PU	Public file.

Refer to PFM Macro Parameter Conventions in this section for further information.

m — File mode. This parameter defines the type of access alternate users may have for semiprivate or public files.

m	Description
W	Read, write, append, purge, and execute (default value).
R	Read and execute.
A	Append.
E	Execute.
M	Read, append, and execute.
N	None.
RM	Same as R for indirect access files.
RA	Same as R for indirect access files.
U	Same as R for indirect access files.
RU	Same as R for indirect access files.

Refer to PFM Macro Parameter Conventions in this section for further information.

pn Address containing one- to seven-character pack name of the auxiliary device on which the file is to be saved.

Entering the display code character zero (left-justified, binary zero fill) in this field overrides the pack name set by the PACKNAM command or macro. This causes the file to saved on your family device.

r Type of auxiliary device on which the file is to be saved (refer to the DEFINE macro).

fo Family option:

IP	The pack name specified by a PACKNAM macro or pn parameter is ignored. PFM accesses your family.

ss Subsystem indicator. This indicator causes automatic subsystem selection when the file is retrieved using the OLD macro or command.

ss	Description
NULL	Null subsystem.
BASIC	BASIC subsystem.
FORT	FORTRAN 5 subsystem.
FTNTS	FORTRAN Extended 4 subsystem.
EXEC	Execute subsystem.
BATCH	Batch subsystem.

Refer to PFM Macro Parameter Conventions in this section for further information.

br Backup requirement. Specifies whether the file is to be backed up on a dump tape when the site dumps permanent files.

br	Description
Y	The file must be backed up on the dump tape.
MD	Backup file on dump tape only if a copy does not exist on alternate storage.
N	The file is not to be backed up on the dump tape.

Refer to PFM Macro Parameter Conventions in this section for further information.

pr Preferred residence. Specifies whether you prefer the file to reside on disk or on alternate storage.

pr	Description
M	The file should reside on alternate storage when it is not being accessed.
N	No preferred residence for the file.
D	The file should reside on disk at all times.
L	The file will be locked to disk. (Special validation is required for this option.)

Refer to PFM Macro Parameter Conventions in this section for further information.

xt — Expiration term or date for password. xt is the address of a word containing either the packed expiration date (format: yymmdd, where yy is the year minus 1970), or the expiration term, the number of days ($<7777_8$) to be added to the current date to determine the expiration date. A value of 7777_8 indicates a nonexpiring password. If this parameter is not specified when a password is established, the system creates a nonexpiring password. You must be validated to assign a permanent file expiration date in order to use this parameter.

al — Security access level of the file. This may be either the name of the access level or the address of a location containing the numerical value of the access level. System OPL common deck COMCVLC can be used to convert access level names to numerical values and vice versa. System OPL common deck COMSMLS is required if you specify this parameter. Using this parameter sets the access level bit (bit 39 of FET+1).

If an access level is specified in the al field of the FET rather than as a macro parameter, your program must also set the access level bit (bit 39 of FET+1).

On a secured system, any access level specified must be valid for your job and for the local and permanent file devices involved.

ac — Alternate CATLIST permission. Specifies whether alternate users may obtain information about the file using CATLIST.

ac	Description
N	Permission denied.
Y	Permission granted.

Refer to PFM Macro Parameter Conventions in this section for further information.

Example 1:

This example saves local file FILE in the permanent file system as a public file name DATA with read permission. The file will be backed up on a dump tape when the site dumps permanent files.

```
          SAVE   FILE,PF,,,PU,R,,,,,Y
            .
            .
            .
FILE      FILEB  BUF,BUFL,(FET=16)
PF        VFD    42/OLDATA,18/0
```

Example 2:

This example performs the same function as example 1.

```
            XTEXT   COMSPFM
             .
             .
             .
            SA1     PF
            SAVE    FILE,X1,,,FCAT,FMOD,,,,,BACK
             .
             .
             .
FCAT        CON     FCPB
FMOD        CON     PTRD
BACK        CON     2
FILE        FILEB   BUF,BUFL,(FET=16)
PF          VFD     42/0LDATA,18/0
```

Example 3:

In this example, local file TAPE1 is saved in the permanent file system as a private file name PTAPE1 with null permission. The subsystem indicator is set to the batch subsystem.

```
            SAVE    F,PF,,,P,N,,,,BATCH
             .
             .
             .
F           BSS     0
TAPE1       FILEB   BUF,BUFL,(FET=16)
PF          VFD     42/0LPTAPE1,18/0
```

GET (002, CCGT)

The GET macro enables the user program to obtain a temporary copy of an indirect access permanent file. If a local file by the same name already exists, it is returned as if the UNLOAD macro had been issued even if the GET is unsuccessful. The new file is set to rewound status. No interlock is provided to prevent other users from obtaining copies of the same file simultaneously.

The local file created by a GET macro call may be an execute-only file, depending on the access mode permitted to the user name associated with the user program. PFM returns the access mode of the file in the mode field (FET+10_8, bits 0 through 5) if the FET is long enough; the user program can examine this field to determine the access mode. If the FET is long enough, the user control word (ucw) is returned to FET+13_8, and the subsystem is returned to FET+17_8, bits 59-57. Refer to PFM Macro Parameter Conventions in this section for a list of access modes and subsystems.

If the access level bit (bit 39 of FET+1) is set, the access level of the file is returned to bits 38 through 36 of FET+4.

Macro format:

Location	Operation	Variable
	GET	addr,pfn,un,pwd,pn,r,fo

addr — Address of the FET; the local file name must be set in FET+0.

pfn — Address containing the name of the file; name the file is given in the permanent file catalog.

un — Address containing the number of the alternate user whose catalog is to be searched for the file specified; if this parameter is specified, the permission mode is that which the user has been permitted for private files or that specified in the catalog for semiprivate and public files.

pwd — Address containing the password of the file; required if un≠0 and the file requires a password.

pn — Address containing one- to seven-character pack name of the auxiliary device from which the file is to be retrieved.

Entering the display code character zero (left-justified, binary zero fill) in this field overrides the pack name set the PACKNAM command or macro. This causes the request to be processed on your family device.

r — Type of auxiliary device on which the indirect access permanent file resides (refer to the DEFINE macro).

fo — Family option:

IP — The pack name specified by a PACKNAM macro or pn parameter is ignored. PFM accesses your family.

PURGE (003, CCPG)

The PURGE macro enables the user program to remove the specified file from the permanent file system. To purge a file in an alternate user's catalog, you must have write permission for the file or the file must be a semiprivate or public file with write mode.

Macro format:

Location	Operation	Variable
	PURGE	addr,un,pwd,pn,r,fo

addr — Address of the FET; the local file name must be set in FET+0. If FET+10_8 contains a permanent file name, that file name is used for the PURGE operation.

un — Address containing the user name of the alternate catalog for the file to be purged.

pwd — Address containing the password of the file; required if un≠0 and the file requires a password.

pn — Address containing one- to seven-character pack name of the auxiliary device on which the file resides.

Entering the display code character zero (left-justified, binary zero fill) in this field overrides the pack name set by the PACKNAM command or macro. This causes the request to be processed on your family device.

r — Type of auxiliary device identified by the pn parameter (refer to the DEFINE macro).

fo — Family option:

IP — The pack name specified by a PACKNAM macro or pn parameter is ignored. PFM accesses your family.

An example of the use of this macro is:

```
      PURGE    F
        .
        .
        .
F     FILEB    BUF,BUFL
```

These instructions purge permanent file F.

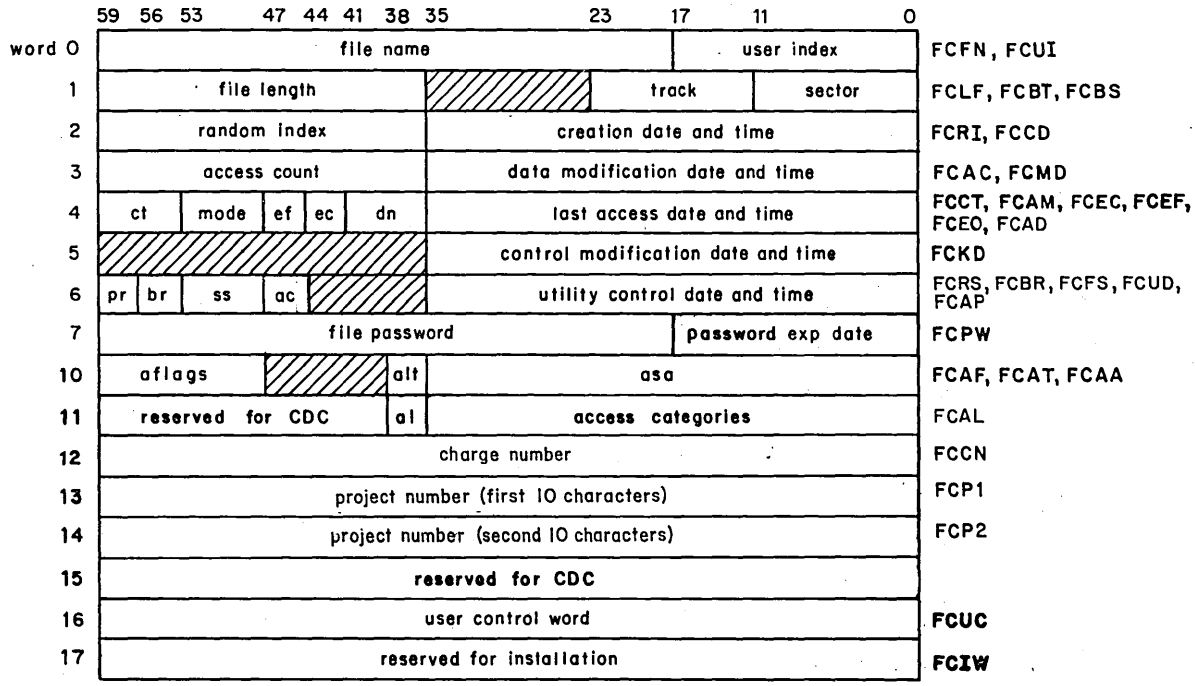
CATLIST (004, CCT)

The CATLIST macro can be used for the following applications:

- To determine the information about one or more of your own permanent files.
- To determine the information about one or more files belonging to a specified alternate user. You will only be allowed to see information about those files which you can access and which the owner has specified you to receive (via the AC = Y parameter).
- To determine the alternate user information for a specified file that an alternate user can access or has accessed in your catalog (permit data).

The first type of application is the return of the permanent file catalog entries of all your files. If a permanent file name is specified in FET+CFPN, the catalog entry of that one file will be returned.

The format of the permanent file catalog entry is as follows:



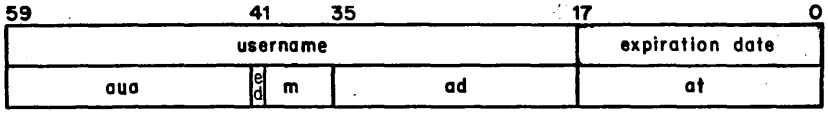
file name	Permanent file name.
user index	User index of file creator.
file length	Length in PRUs of the file.
track	Beginning track of file.
sector	Beginning sector of file (4xxx for a direct access file).
random index	Random disk address of permit sector.

creation date and time	Date and time (yymmddhhmmss in octal) when this file was first entered on the permanent file system. The year (yy) is the calendar year minus 1970.
access count	Count of accesses to file.
data modification date and time	Date and time (yymmddhhmmss in octal) when the data in this file was last modified. The year (yy) is the calendar year minus 1970. For direct access files this field is updated when the file is attached in a modifiable mode, whether or not the file is actually modified.
ct	File category (refer to PFM Macro Parameter Conventions for further information).
mode	File mode (refer to PFM Macro Parameter Conventions for further information).
ef	Error flag: 0 No error. 1 EOI changed by recovery.
ec	Error code: 0 No error. 1 Error in data for file. 2 Error in permit entries for file. 3 Error in data and permit entries for file. 4 File length error. 5-7 Reserved.
dn	Device number (0 through 77_8); each device within a family of permanent file devices is identified by a device number.
last access date and time	Date and time (yymmddhhmmss in octal) when this file was last accessed. The year (yy) is the calendar year minus 1970.
control modification date and time	Date and time (yymmddhhmmss in octal) when the control information (catalog entry and permit record data) for this file was last updated. This corresponds to the last time a CHANGE or PERMIT command or macro was processed for the file. The year (yy) is the calendar year minus 1970.
pr	Preferred residence for file (refer to PFM Macro Parameter Conventions in this section for further information).
br	Backup requirement for file (refer to PFM Macro Parameter Conventions in this section for further information).
ss	Subsystem code for this file (refer to PFM Macro Parameter Conventions in this section for further information).
ac	Alternate CATLIST permission for this file (refer to PFM Macro Parameter Conventions in this section for further information).

utility control date and time	Date and time (yymmddhhmmss in octal) used to determine this file's candidacy for being dumped by permanent file utilities. The year (yy) is the calendar year minus 1970.
file password	Optional password.
password exp date	The date on which the password expires (yymmdd in octal). The year (yy) is the calendar year minus 1970.
aflags	Alternate storage flags; used with the asa field and meaningful only if the file resides on an alternate storage device (as indicated by the alt field); refer to the NOS 2 Analysis Handbook for a complete description of this field.
alt	Alternate storage type: 0 No alternate storage copy of the file exists. 1 An alternate storage copy of the file resides on MSF.
asa	Alternate storage address; meaningful only if an alternate storage copy of the file exists (as indicated by the alt field); if asa is 0, the file does not reside on alternate storage; refer to the NOS 2 Analysis Handbook for a complete description of this field.
al	File's security access level.
access categories	Each of bits 31 through 0, if set, indicates that the corresponding category is in the security access category set for the file. The system OPL common deck COMCVLC can be used to convert an access category value to an access category name.
charge number	Charge number to which the file will be billed.
project number	Project number to which the file will be billed.
user control word	User control information (FET+13_8).

The second type of application is a CATLIST of another user's files. The system returns the permanent file catalog entries of all the user's files you are permitted to access and that are listable by another user. (A file is listable by way of CATLIST only if the owner has specified an alternate CATLIST permission [AC=Y].) If a permanent file name is specified in FET+CFPN, the catalog entry of the specified file will be returned (provided the file is listable by CATLIST and that you are permitted to access it). The following fields are cleared before the catalog entries are returned: user index, access level, access category set, password, and password expiration date.

The third type of application is a CATLIST of PERMIT data. All alternate user access entries (permit entries) for a specified file are returned. The format of an alternate user access entry is illustrated.



username — Alternate user name.

expiration date — The date on which the permit expires [yymmdd in octal; the year (yy) is the calendar year minus 1970].

aua — Number of accesses the alternate user has made to the file (hhmmss in octal).

ed — Bit 41; if set, indicates that an expiration date is present in bits 17-0 of the first word of the entry.

m — Permission mode (bit 40 set if this was an accounting permit and was not created by a PERMIT command or macro; bit 40 clear indicates an explicit permit set by PERMIT command or macro; bits 39 through 36 are same as mode described in first type of request).

ad — The last date the alternate user accessed the file [yymmdd in octal; the year (yy) is the calendar year minus 1970].

at — The time of day the alternate user last accessed the file (hhmmss in octal).

Macro format:†

Location	Operation	Variable
	CATLIST	addr,pfn,un,m,pn,r,fo,dn

addr — Address of the FET for the CATLIST function.

pfn — Address containing the file name; if this parameter is omitted, CATLIST information for all files in your permanent file catalog (or in the catalog specified by un) is returned.

un — Address containing the name of the alternate user catalog to be searched for the catalog information. If un is omitted, your own catalog is searched.

m — If this parameter is the letter M, alternate user access information for file pfn is returned. If this parameter is specified, the pfn is required and un is ignored.

†This macro is not available in SYSTEXT. The user program must specify systems text PSSTEXT or call common deck COMCMAC (refer to appendix F).

pn — Address containing one- to seven-character pack name of the auxiliary device that contains catalog information for all users with information on that device.

Entering the display code character zero (left-justified, binary zero fill) in this field overrides the pack name set by the PACKNAM command or macro. This causes the request to be processed on your family device.

r — Type of auxiliary device identified by the pn parameter (refer to the DEFINE macro).

fo — Family option:

IP — The pack name specified by a PACKNAM macro or pn parameter is ignored. PFM accesses your family.

dn — Request CATLIST of files on device number dn (FET+11$_8$, bits 17 through 12). Any value may be specified.

If the status returned in FET+0 (addr parameter of the CATLIST macro), bits 9 through 0, is 0033 (EOF encountered), the user program should reissue the CATLIST macro after the buffer of entries has been processed (refer to example 1). The user program should continue this until a status of 1033 (EOI encountered) is returned. CATLIST uses the current random index field (FET+6, bits 59 through 30) to keep track of its position for continued calls. If the user program changes this field, the results of a CATLIST request may be undefined.

Information is placed in the buffer starting at IN until (IN=LIMIT-1), at which time buffer full (0033) status is set in FET+0. PFM does not process the buffer circularly; therefore, the user program must reset (IN=OUT=FIRST) before reissuing the CATLIST macro.

The following examples assume that PSSTEXT is specified at assembly.

Example 1:

The following program creates a binary file named F with permanent file catalog entries for all files in your catalog. (The system OPL must be attached to assemble this program.)

```
            IDENT     CAT,FWA
            ABS
            ENTRY     START
            SYSCOM    B1

OPL         XTEXT     COMSPFM
            ORG       112B
FWA         BSS       0
BUFL        EQU       1001B
F           FILEB     BUFF,BUFL
G           FILEB     BUFG,BUFL,FET=10D

START       SB1       1

STA1        CATLIST   G

STA2        READW     G,WBUF,NWCE       TRANSFER CATALOG ENTRY
            NG        X1,STA3           IF BUFFER EMPTY
            WRITEW    F,WBUF,NWCE
            EQ        STA2              TRANSFER NEXT CATALOG ENTRY

STA3        SX1       X1+B1
            NG        X1,STA4           IF EOI (CATLIST COMPLETE)
            SA1       G+1               RESET FET POINTERS
            SX7       X1
            SA7       A1+B1             SET IN=FIRST
            SA7       A7+B1             SET OUT=FIRST
            EQ        STA1              GET NEXT BUFFER LOAD

STA4        WRITER    F
            ENDRUN

BUFF        BSS       BUFL
BUFG        BSS       BUFL
WBUF        BSS       NWCE
OPL         XTEXT     COMCCIO
OPL         XTEXT     COMCPFM
OPL         XTEXT     COMCRDW
OPL         XTEXT     COMCSYS
OPL         XTEXT     COMCWTW

            END
```

Example 2:

The following example returns all access information for file DATA013.

```
          CATLIST   F,PF,,M
           .
           .
           .
F         FILEB     BUF,BUFL,(FET=10D)
PF        VFD       42/0LDATA013,18/0
```

Example 3:

This example returns a list of all files that you can access by way of CATLIST in the catalog of user USERABC. [A file is listable by way of CATLIST only if the owner has specified an alternate CATLIST permission (AC=Y).]

```
          CATLIST   F,,AUN
           .
           .
           .
F         FILEB     BUF,BUFL,(FET=10D)
AUN       VFD       42/0LUSERABC,18/0
```

PERMIT (005, CCPM)

The PERMIT macro enables your program to explicitly permit another user to access a private or semiprivate file in your permanent file catalog. If you wish to remove permission previously granted, you should select the negate (null) mode.

Macro format:

Location	Operation	Variable
	PERMIT	addr,pfn,un,m,pn,r,fo,,xt

addr — Address of the FET; the local file name must be set in FET+0.

pfn — Address containing the name of the file; name of the file is given in the permanent file catalog.

un — Address containing the user name of the user being granted permission.

m — Mode of permission being granted; if no mode is specified, W (write) is assumed.

m	Description
W	Read, write, modify, append, update, purge, and execute (default value).
R	Read and execute.
A	Append.
E	Execute.
N	None (negate previous permissions).
M	Modify, append, update, read, and/or execute.

m	Description
RM	Read and/or execute, allow simultaneous access in modify, append, or update mode.
RA	Read and/or execute, allow simultaneous access in append mode.
U	Update.
RU	Read and/or execute, allow simultaneous access in update mode.

Refer to PFM Macro Parameter Conventions in this section for further information.

pn — Address containing one- to seven-character pack name of the auxiliary device on which the specified file resides.

Entering the display code character zero (left-justified, binary zero fill) in this field overrides the pack name set by the PACKNAM command or macro. This causes the request to be processed on your family device.

r — Type of auxiliary device identified by the pn parameter (refer to the DEFINE macro).

fo — Family option:

IP — The pack name specified by a PACKNAM macro or pn parameter is ignored. PFM accesses your family.

xt — Expiration term or date for permit. xt is the address of a word containing either the packed expiration date (format: yymmdd, where yy is the year minus 1970), or the expiration term, the number of days ($<7777_8$) to be added to the current date to determine the expiration date. A value of 7777_8 indicates a nonexpiring password. If this parameter is not specified, the system sets a site-defined default permit expiration date. You must be authorized to assign a permanent file expiration date in order to use this parameter; also, your site may choose to limit the maximum expiration term you may specify.

An example of the use of this macro is:

```
        PERMIT F,PF,AUN,A
          .
          .
          .
F       FILEB  BUF,BUFL,(FET=10D)
PF      VFD    42/0LDATA012,18/0
AUN     VFD    42/0LUSERABC,18/0
```

This allows user USERABC to have append permission to file DATA012.

REPLACE (006, CCRP)

The REPLACE macro enables the user program to place a copy of the specified local mass storage file in the permanent file system as an indirect access file. If an indirect access permanent file with the specified name already exists in the catalog, that file will be replaced by the new file placed in the catalog as the same type of file. If the file does not exist in the catalog, a new indirect access permanent file is created as if a SAVE request had been made. Permission information and alternate user access data are not lost when the file is replaced.

Macro format:

Location	Operation	Variable
	REPLACE	addr,pfn,un,pwd,ucw,pn,r,fo

addr — Address of the FET; local file name must be set in FET+0.

pfn — Address containing the name of the permanent file to be replaced in the permanent file catalog.

un — Address containing the name of the alternate user's catalog where the file resides.

pwd — Address containing the password of the file being replaced (this may be required on an alternate user access).

ucw — Address containing user control word to place with the file; bit 59 of FET+13_8 must be set to retain the word with the file in the catalog.

pn — Address containing one- to seven-character pack name of the auxiliary device on which the file is to be placed.

Entering the display code character zero (left-justified, binary zero fill) in this field overrides the pack name set by the PACKNAM command or macro. This causes the file to be replaced on your family device.

r — Type of auxiliary device identified by the pn parameter (refer to the DEFINE macro).

fo — Family option:

IP — The pack name specified by a PACKNAM macro or pn parameter is ignored. PFM accesses your family.

APPEND (007, CCAP)

The APPEND macro enables a user program to add the contents of the specified local mass storage files to the end of the specified indirect access permanent file. The logical structure of the two files is retained; that is, EORs and EOFs are appended as well as data. If the working file is appended to a file in an alternate user's catalog, a password must be supplied if one is required.

Macro format:

Location	Operation	Variable
	APPEND	addr,pfn,un,pwd,pn,r,fo

addr — Address of the FET; name of the local file to be appended must be set in FET+0.

pfn — Address containing the name of the permanent file to which information is to be appended.

un — Address containing the name of the alternate user whose catalog contains the permanent file.

pwd — Address containing the password of the file to which information is to be appended. This must be specified if un≠0 and the file requires a password.

pn — Address containing one- to seven-character pack name of the auxiliary device on which the specified permanent file resides.

Entering the display code character zero (left-justified, binary zero fill) in this field overrides the pack name set by the PACKNAM command or macro. This causes the request to be processed on your family device.

r — Type of auxiliary device identified by the pn parameter (refer to the DEFINE macro).

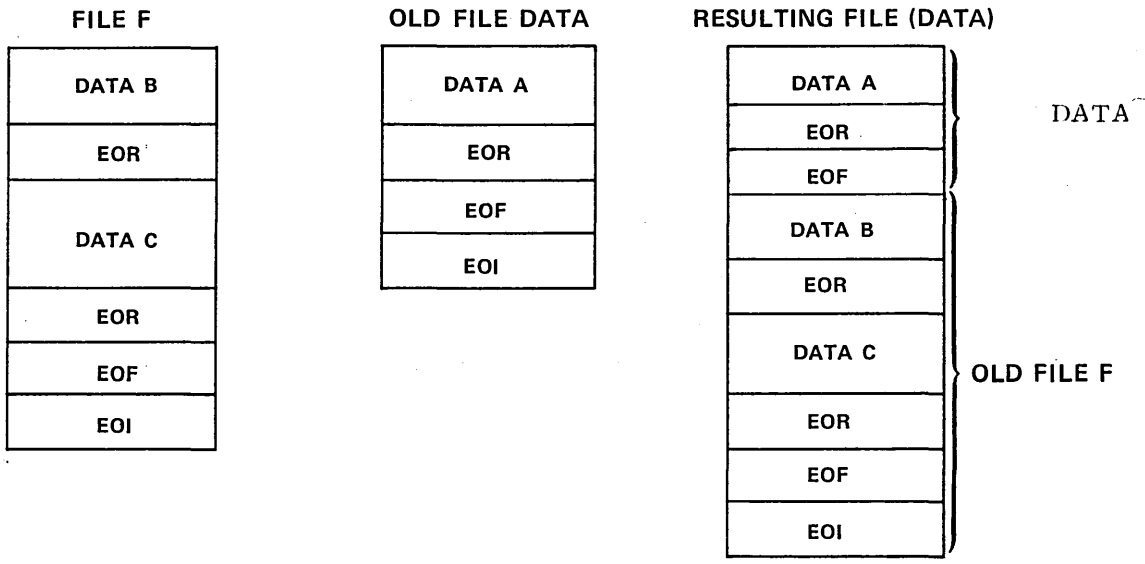
fo — Family option:

IP — The pack name specified by a PACKNAM macro or pn parameter is ignored. PFM accesses your family.

An example of the use of this macro is:†

```
        APPEND  F
          .
          .
          .
    F   FILEB   BUF,BUFL,(PFN=DATA)
          .
          .
```

The structure of file DATA after the APPEND macro is issued is as follows.



†The permanent file name can be set by FET creation macros as well as being set in the permanent file macros.

DEFINE (010, CCDF)

The DEFINE macro enables the user program to create a direct access permanent file. If the user program attempts to create a direct access file with the same name as a local file, the local file must reside on a legal permanent file device for your user name. If a local file with the same name does not exist, an empty file is created on an appropriate device.

Macro format:

Location	Operation	Variable
	DEFINE	addr,pfn,pwd,ucw,r,ct,m,pn,s,fo,br,pr,,xt,al,ac

addr — Address of the FET. FET+0 must contain the local file name of the file to be made a direct access file (name used when the job accesses the file).

pfn — Address containing the name of the permanent file entered in the catalog. If pfn=0, the contents of FET+0 is used for the permanent file name.

pwd — If pwd is specified, the password is retained with the file; all alternate users must supply this password when attaching file.

ucw — Address containing user control information, bit 59 of FET+13_8 must be set to retain this data. If this parameter is supplied, this data is retained in the catalog entry for the file and returned to FET+13_8 when the file is attached.

r† — Specifies the type of family or auxiliary device on which the permanent file resides or is to reside; r can be DBn, DDn, DE, DIn, DJn, DKn, DLn, DMn, DQn, DP, DW, or DV as described for Equipment Codes in appendix E. If r is not one of these values, the value specified represents the address that contains the display code equivalent, left-justified. If the device is a multiunit type device (DBn, DDn, DIn, DJn, DKn, DLn, DMn, or DQn), n is the number of physical units which the installation has combined into one logical unit; n is converted from display code to octal and placed in FET+14_8, bits 11 through 0. Also, if r is DB, DD, DI, DJ, DK, DM, or DQ but n is omitted, the unit count is assumed to be 1. If r is not specified, the system default device type (or the device type specified in the last PACKNAM macro or command) is used.

ct — File category:

ct	Description
P	Private file (default value).
S	Semiprivate.
PU	Public file.

Refer to PFM Macro Parameter Conventions in this section for further information.

†If the file later resides on MSF and then is staged back to disk, it may not reside on the device type specified by the r parameter.

m — File mode. This parameter defines the type of access alternate users may have for semiprivate or public files.

m	Description
W	Read, write, modify, append, update, purge, and execute (default value).
R	Read and execute.
A	Append.
E	Execute.
N	None.
M	Modify, append, update, read, and/or execute.
RM	Read and/or execute, allow simultaneous access in modify, append, or update mode.
RA	Read and/or execute, allow simultaneous access in append mode.
U	Update.
RU	Read and/or execute, allow simultaneous access in update mode.

Refer to PFM Macro Parameter Conventions in this section for further information.

pn — Address containing the one- to seven-character pack name of the auxiliary device on which the direct access file is to reside.

Entering the display code character zero (left-justified, binary zero fill) in this field overrides the pack name set by the PACKNAM command or macro. This causes the file to be defined on your family device.

s — Address containing the number of PRUs desired for the direct access file. The number of PRUs is in octal, right-justified. DEFINE places this value in FET+11$_8$ bits 23 through 0 (CFOU).

fo — Family option:

IP	The pack name specified by a PACKNAM macro or pn parameter is ignored. PFM accesses your family.

br — Backup requirement. Specifies whether the file is to be backed up on a dump tape when the site dumps permanent files.

br	Description
Y	The file must be backed up on the dump tape.
MD	Backup file on dump tape only if it does not reside on alternate storage.
N	The file is not to be backed up on the dump tape.

Refer to PFM Macro Parameter Conventions in this section for further information.

pr Preferred residence. Specifies whether you prefer the file to reside on disk or on alternate storage.

pr	Description
M	The file should reside on alternate storage when it is not being accessed.
N	No preferred residence for the file.
D	The file should reside on disk at all times.
L	The file will be locked to disk. (Special validation is required for this option.)

Refer to PFM Macro Parameter Conventions in this section for further information.

xt Expiration term or date for password. xt is the address of a word containing either the packed expiration date (format: yymmdd, where yy is the year minus 1970), or the expiration term, the number of days ($<7777_8$) to be added to the current date to determine the expiration date. A value of 7777_8 indicates a nonexpiring password. If this parameter is not specified when a password is established, the system creates a nonexpiring password. You must be validated to assign a permanent file expiration date in order to use this parameter; also, your site may choose to limit the maximum expiration term you may specify.

al Security access level of the file. This may be either the name of the access level or the address of a location containing the numerical value of the access level. System OPL common deck COMCVLC can be used to convert access level names to numerical values and vice versa. System OPL common deck COMSMLS is required if you specify this parameter. Using this parameter sets the access level bit (bit 39 of FET+1).

If an access level is specified in the al field of the FET rather than as a macro parameter, your program must also set the access level bit (bit 39 of FET+1).

On a secured system, any access level specified must be valid for your job and for the local and permanent file devices involved.

ac Alternate CATLIST permission. Specifies whether alternate users may obtain information about the file using CATLIST.

ac	Description
N	Permission denied.
Y	Permission granted.

Refer to PFM Macro Parameter Conventions in this section for further information.

If lfn does not exist at the time the DEFINE request is issued, the device on which pfn resides depends on the r and s parameters.

r	s	Residency
Specified	Not specified	The file resides on the device of type r with the most space available.
Specified	Specified	The file resides on the device of type r with the most space available, provided that device has as many PRUs available as specified by the s parameter.
Not specified	Specified	The file resides on the device with the most space available, provided that device has as many PRUs available as specified by the s parameter.
Not specified	Not specified	The file resides on the device with the most space available.

Example 1:

This defines file F as a read-only public file on a DI device. The name of the entry in the permanent file catalog is DATA. (It is assumed that the file did not exist before the DEFINE request was issued.)

```
          DEFINE  F,PF,,,DI,PU,R
                                .
                                .
                                .
F         FILEB   BUF,BUFL, (FET=9)
PF        VFD     42/0LDATA,18/0
```

Example 2:

The same operation could be accomplished by the following sequence of instructions.

```
          SA1     PF
          DEFINE  F,X1,,,RES,FCAT,R
           .
           .
           .
F         FILEB   BUF,BUFL,(FET=9)
PF        VFD     42/0LDATA,18/0
RES       VFD     12/0LDI,48/0
FCAT      CON     FCPU
OPL       XTEXT   COMSPFM
```

Example 3:

The following sequence defines file DATA. The file will not be backed up on the archive tape if it is residing on MSF, and its preferred residence is MSF.

```
          DEFINE  P,PF,,,,,,,,,MD,M
           .
           .
           .
F         FILEB   BUF,BUFL,(FET=16D)
PF        VFD     42/0LDATA,18/0
```

ATTACH (011, CCAT)

The ATTACH macro enables your program to attach the specified direct access file to your job.

A read/write interlock is provided to ensure that only one user at a time accesses the file in write mode. Several users may access the file in read mode simultaneously. You should return the file as soon as possible to enable other users to access the file. If a local file by the same name already exists, it is returned as if the UNLOAD macro had been issued even if the ATTACH is unsuccessful.

If the FET is long enough, the user control word for the file is returned to FET+13_8.

If the access level bit (bit 39 of FET+1) is set, the access level of the file is returned to bits 38 through 36 of FET+4.

Macro format:

Location	Operation	Variable
	ATTACH	addr,pfn,un,pwd,m,pn,r,fo,sr,rt

addr — Address of the FET. FET+0 must contain the local file name of the file while attached to the job.

pfn — Address containing the name of the file in the permanent file catalog. If pfn is 0, the contents of FET+0 is used for the permanent file name.

un — Address containing name of the alternate user catalog on which the file resides.

pwd — Address containing file password; entered only if un is not 0 and the file requires a password.

m — Mode of access desired:

m	Description
W	Read, write, modify, append, update, purge, and/or execute (default value).
R	Read and/or execute.
A	Append, update, read, and/or execute.
E	Execute.
N	None.
M	Modify, append, update, read, and/or execute.
RM	Read and/or execute, allow simultaneous access in modify, append, or update mode.
RA	Read and/or execute, allow simultaneous access in append mode.
U	Update, read, and/or execute.
RU	Read and/or execute, allow simultaneous access in update mode.

Refer to PFM Macro Parameter Conventions in this section for further information.

pn — Address containing one- to seven-character pack name of the auxiliary device on which the specified permanent file resides (refer to the DEFINE macro).

Entering the display code character zero (left-justified, binary zero fill) in this field overrides the pack name set by the PACKNAM command or macro. This causes the file to be processed on your family device.

r — Type of auxiliary device identified by the pn parameter (refer to the DEFINE macro).

fo — Family option:

- IP — The pack name specified by a PACKNAM macro or pn parameter is ignored. PFM accesses your family.

sr — Special request subfunction:

sr	Description
FA†	The file being attached must be a fast attach permanent file.
IE†	Ignore error idle status.
NF	The file being attached may not be a fast attach permanent file.

rt — Real-time mode. If this parameter is specified (any value may be used), the job step is not delayed for either of the following conditions:

- The file must be staged to disk from MSF.
- A permanent file utility is currently active, preventing the ATTACH processing.

If neither of these conditions exists, the rt parameter has no meaning. If one of these conditions exists and the rt parameter is not specified, the job step is delayed until the operation is complete. If one of these conditions exists and the rt parameter is specified, the file is not attached; instead, the operation to ensure that the file resides on disk is initiated. To complete the attach, another ATTACH macro with or without the rt parameter should be issued and the FET status checked to ensure that the file was accessed. The function of the rt parameter is also accomplished by setting either the up or rt bit in the FET.

If a file is busy at the time the user program issues an ATTACH macro for the file, the system aborts the request. The user program can bypass the abort by specifying error processing (the ep bit, bit 44 of FET+1). If ep is set and the file is busy, the system returns control to the user program. The user program may then suspend the job by issuing a ROLLOUT macro. Normally, when a program issues a ROLLOUT macro to roll out a job subject to time/event dependencies, the program must include an address specifying the time period and/or event. However, when file busy status is returned, PFM sets up a time/event entry for the user program, specifying a rollout time period of 360_8 seconds. If the file becomes available before the time period expires, the job is rolled in at that time. The user program should then attempt the ATTACH once again.

Refer to the ATTACH command in Volume 3, System Commands, for a description of the current access modes that result when you attempt to attach an active file.

†The calling program must have an SSJ= entry point.

CHANGE (012, CCCG)

The CHANGE macro allows you to alter the attributes of one of your direct or indirect access permanent files without having to recreate the file. The pn and r parameters cannot be used to specify a new auxiliary device. They are used only to specify the auxiliary device on which ofn resides. Although all attributes that may be specified by parameters on the macro may also be specified using the appropriate fields in the FET, the file category, file access mode, file subsystem, and user control word must each have the top bit of the field set if they are to be specified in the FET for this request.

Macro format:

Location	Operation	Variable
	CHANGE	addr,ofn,nfn,pwd,ucw,ct,m,pn,r,fo,sr,ss,br,pr,xt,ac

addr — Address of the FET; the local file name must be set in FET+0.

ofn — Address containing the old permanent file name. If ofn is not specified, the contents of FET+0 is used instead.

nfn — Address containing the new permanent file name. If nfn is not specified, no name change takes place.

pwd — Address containing new password; replaces the old password or adds a new password if there was none before. If pwd is omitted, the contents of FET+18_8 bits 59 through 18 are used as the new password. If you do not wish to change the current password, either FET+12_8 must be set to 42/77777777777777B,18/0 or the contents of location pwd must be set to 42/77777777777777B,18/0.

ucw — Address of word containing user control information; bit 59 of the word must be set if this information is to be used. If ucw is not specified, the contents of FET+13_8 are used instead; bit 59 of FET+13_8 must be set in that case.

ct — File category:

ct	Value (Octal)	Description
P	40	Private file.
S	41	Semiprivate.
PU	42	Public file.

Refer to PFM Macro Parameter Conventions in this section for further information.

m — File mode. This parameter defines the type of access alternate users may have for semiprivate or public files.

m	Value (Octal)	Description
W	40	Read, write, modify, append, update, purge, and execute (default value).
R	41	Read and execute.
A	42	Append.
E	43	Execute.
N	44	None.
M	45	Modify, append, update, read, and/or execute.
RM	46	Read and/or execute, allow simultaneous access in modify, append, or update mode.
RA	47	Read and/or execute, allow simultaneous access in append mode.
U	50	Update.
RU	51	Read and/or execute, allow simultaneous access in update mode.

Refer to PFM Macro Parameter Conventions in this section for further information.

pn — Address containing the one- to seven-character pack name of the auxiliary device on which the file resides.

Entering the display code character zero (left-justified, binary zero fill) in this field overrides the pack name set the the PACKNAM command or macro. This causes the request to be processed on your family device.

r — Address containing the type of auxiliary device identified by the pn parameter (refer to the DEFINE macro).

fo — Family option:

IP — The pack name specified by a PACKNAM macro or pn parameter is ignored. PFM accesses your family.

sr — Special request subfunction (only one may be specified):

CE — Clear file error code in the catalog entry (refer to the CATLIST macro for a description of the catalog entry).

CP — Reset the charge number and project number for the file to your job's current charge and project number.

ss — Subsystem indicator. This indicator causes automatic subsystem selection when the file is retrieved using the OLD macro or command.

ss	Description
NULL	Null system.
BASIC	BASIC subsystem.
FORT	FORTRAN 5 subsystem.
FTNTS	FORTRAN Extended 4 subsystem.
EXEC	Execute subsystem.
BATCH	Batch subsystem.

Refer to PFM Macro Parameter Conventions in this section for further information.

br Backup requirement. Specifies whether the file is to be backed up on a dump tape when the site dumps permanent files.

br	Description
Y	The file must be backed up on the dump tape.
MD	Back up file on dump tape only if it does not reside on alternate storage.
N	The file is not to be backed up on the dump tape.

If the backup requirement is not to be changed, the br field of the FET must be 0.

Refer to PFM Macro Parameter Conventions in this section for further information.

pr Preferred residence. Specifies whether you prefer the file to reside on disk or on alternate storage.

pr	Description
M	The file should reside on alternate storage when it is not being accessed.
N	No preferred residence for the file.
D	The file should reside on disk at all times.
L	The file will be locked to disk (special validation is required for this option).

If the preferred residence is not to be changed, the pr field of the FET must be 0.

Refer to PFM Macro Parameter Conventions in this section for further information.

xt Expiration term or date for password. xt is the address of a word containing either the packed expiration date (format: yymmdd, where yy is the year minus 1970), or the expiration term, the number of days ($<7777_8$) to be added to the current date to determine the expiration date. A value of 7777_8 indicates a nonexpiring password. If this parameter is not specified when a password is established, the system creates a nonexpiring password. You must be validated to assign a permanent file expiration date in order to use this parameter; also, your site may choose to limit the maximum expiration term you may specify. This parameter may not be specified without also specifying the pwd parameter unless the file already has a password.

ac Alternate CATLIST permission. Specifies whether alternate users may obtain information about the file using CATLIST.

ac	Description
N	Permission denied.
Y	Permission granted.

Refer to PFM Macro Parameter Conventions in this section for further information.

Example 1:

This changes file DATA on auxiliary device PACK1 to FILE with read-only access.

```
            CHANGE F,PF,NF,PW,,,R,PKN
             .
             .
             .
F           FILEB  BUF,BUFL,(FET=14D)
PF          VFD    42/0LDATA,18/0
NF          VFD    42/0LFILE,18/0
PKN         VFD    42/0LPACK1,18/0
PW          VFD    42/-0,18/0
```

Example 2:

This example changes the backup requirement of file DATA to MD (backup file on dump tape only if it does not reside on MSF) and the preferred residence to M (MSF).

```
            CHANGE DATA,,,PW,,,,,,,,,MD,M
             .
             .
             .
DATA        FILEB  BUF,BUFL,(FET=16D)
PW          VFD    42/-0,18/0
```

OLD (021, CCOD)

The OLD macro enables the user program to obtain a temporary copy of an indirect access permanent file and make it the primary file. If a local file by the same name already exists, it is returned as if the UNLOAD macro had been issued even if the OLD is unsuccessful. If a primary file exists (with a different local file name) before the OLD request, it is converted to a local (LOFT) file. The new primary file is set to rewound status. No interlock is provided to prevent other users from obtaining copies of the same file simultaneously.

The OLD macro also sets the subsystem index associated with the file.

The local file created by an OLD macro call may be an execute-only file, depending on the access mode permitted to the user name associated with the user program. PFM returns the access mode of the file in the mode field (FET+10_8, bits 0 through 5) if the FET is long enough; the user program can examine this field to determine the access mode. If the FET is long enough, the user control word (ucw) is returned to FET+13_8, and the subsystem is returned to FET+17_8, bits 59-57. Refer to PFM Macro Parameter Conventions in this section for a list of access modes and subsystems.

If the access level bit (bit 39 of FET+1) is set, the access level of the file is returned to bits 38 through 36 of FET+4.

Macro format:

Location	Operation	Variable
	OLD	addr,pfn,un,pwd,pn,r,fo

addr — Address of the FET; the local file name must be set in FET+0.

pfn — Address containing the name of the file; name the file is given in the permanent file catalog.

un — Address containing the name of the alternate user whose catalog is to be searched for the file specified; if this parameter is specified, the permission mode is that which the user has been permitted for private files or that specified in the catalog for semiprivate and public files.

pwd — Address containing the password of the file; required if un≠0 and the file requires a password.

pn — Address containing one- to seven-character pack name of the auxiliary device from which the file is to be retrieved.

Entering the display code character zero (left-justified, binary zero fill) in this field overrides the packname set by the PACKNAM command or macro. This causes the file to be saved on the family device of the user name associated with the user program.

r — Type of auxiliary device on which the indirect access permanent file resides (refer to the DEFINE macro).

fo — Family option:

IP — The pack name specified by a PACKNAM macro or pn parameter is ignored. PFM accesses your family.

SETPFAC (022,CCAC)

The SETPFAC macro enables the user program to change the security access category set of a permanent file.

If this function is to be performed on a direct access file, the file must not be attached to any job.

Macro format:

Location	Operation	Variable
	SETPFAC	addr,pfn,cat,pn,r

addr — Address of the FET for the file.

pfn — Address containing the name (left-justified, binary zero fill) of the permanent file whose access category set is to be changed. If this parameter is omitted, the contents of FET+0 are used for the permanent file name.

cat — Address containing the new security access category set for the file. Each of bits 0 through 31 that is set in this word sets the corresponding security access category. The access categories specified must be valid for your job (refer to Security Access Categories in section 1). The system OPL common deck COMCVLC can be used to convert access category names to numerical values and vice versa.

pn — Address containing the one- to seven-character pack name of the auxiliary device on which the file resides.

r — Type of auxiliary device identified by the pn parameter (refer to the DEFINE macro).

SETPFAL (023,CCAL)

The SETPFAL macro enables the user program to change the security access level of a permanent file. This macro requires the system OPL common deck COMSMLS.

If this function is to be performed on a direct access file, the file must not be attached to any job.

Macro format:

Location	Operation	Variable
	SETPFAL	addr,pfn,al,pn,r

addr — Address of the FET for the file.

pfn — Address containing the name (left-justified, binary zero fill) of the permanent file whose access level is to be changed. If this parameter is omitted, the contents of FET+0 are used for the permanent file name.

al — The security access level to be assigned to the file. al can be either the name of the security access level, or the address of a word containing the numerical value of a security access level, right-justified with binary zero fill. To convert an access level name to a numerical value (or vice versa), use system OPL common deck COMCVLC. The access level specified must be valid for your job (refer to Security Access Levels in section 1) and valid for the device on which the file resides.

pn — Address containing the one- to seven-character pack name of the auxiliary device on which the file resides.

r — Type of auxiliary device identified by the pn parameter (refer to the DEFINE macro).

The following examples show two different ways to specify an access level using the SETPFAL macro.

Example 1:

In this example, the file's access level is specified in the macro call and therefore must be an access level name.

```
          .
          .
          .
OPL       XTEXT   COMSMLS
OPL       XTEXT   COMCPFM
TEST      FILEB   BUF,101B,(FET=13D)
BUF       BSS     101B
          .
          .
          .
          SETPFAL TEST,,LVL3
          .
          .
          .
```

Example 2:

In this example, the file's access level is specified in a separate word and therefore must be a numerical value.

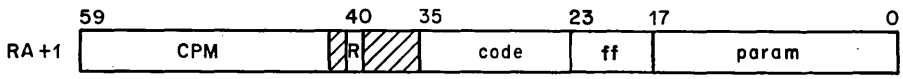
```
          .
          .
          .
OPL       XTEXT   COMSMLS
OPL       XTEXT   COMCPFM
TEST      FILEB   BUF,101B,(FET=13D)
BUF       BSS     101B
LEVEL     CON     3
          .
          .
          .
          SETPFAL TEST,,LEVEL
          .
          .
          .
```

CONTROL POINT MANAGER 6

Control point manager (CPM) enables your program to alter or interrogate parameters in your job's control point area.

Unless otherwise noted, all errors encountered by CPM cause the job step to be aborted; no user error processing is available.

The format of the call to CPM is:



R	Auto recall bit.
code	CPM function code.
ff	Subfunction code for functions 003 and 017.
param	Parameter for the function.

Common decks required for absolute assemblies by the functions processed by CPM are COMCCPM and COMCSYS. For relocatable assemblies, these decks are satisfied by default from the library SYSLIB.

SETPR (001)

The SETPR macro alters the CPU priority of a job. The CPU priority controls the assignment of the CPU to executing jobs that are in memory. If the CPU priority is lower than that of other jobs, the job is assigned to the CPU only when jobs of a higher priority do not need it. Each service class is assigned an initial CPU priority at system initialization. A user job cannot exceed this priority. If the user program requests a priority that exceeds this value or 70_8 (maximum CPU priority), the maximum for the current service class is used.

Macro format:†

Location	Operation	Variable
	SETPR	n

n	CPU priority; ($1 \leq n \leq 70_8$); if n exceeds that for the current service class, it is reduced to that value.

†This macro is not available in SYSTEXT. The user program must specify systems text PSSTEXT or call common deck COMCCMD or COMCMAC (refer to appendix F).

MODE (002)

The MODE macro defines the error conditions that cause the system to exit from normal processing and, on CYBER 180-class models, changes the instruction stack purging status of the central processor (refer to the exchange package description in appendix E).

When the error specified in a MODE macro occurs, the system sets the appropriate error flag and exits from normal processing to perform any error processing required. If an error occurs for which the user program did not select the exit mode processing, the system ignores the error and continues normal processing.

Macro format:†

Location	Operation	Variable
	MODE	m,n,p

m — CPU program error exit mode.

n — This parameter is included for compatibility with earlier versions of NOS. The value supplied is ignored.

p — Instruction stack purging (CYBER 180-class models only):

- 0 Normal instruction stack purging.
- 1 Extended instruction stack purging.

The processor will exit when this condition occurs	If m has one of these octal values
Address out of range (automatically selected for model 176)	1, 3, 5, 7, 11, 13, 15, 17††
Operand out of range (overflow only on model 176)	2, 3, 6, 7, 12, 13, 16, 17
Indefinite operand	4, 5, 6, 7, 14, 15, 16, 17
Underflow (model 176 only)	10, 11, 12, 13, 14, 15, 16, 17

The maximum value and default on all models except model 176 is 7, indicating that the processor exits on all conditions. The maximum value on model 176 is 17_8, indicating that the processor exits on all conditions. The default value on model 176 is 7, indicating that the processor exits on all conditions except underflow.

†This macro is not available in SYSTEXT. The user program must specify systems text PSSTEXT or call common deck COMCCMD or COMCMAC (refer to appendix F).

††On a model 176, the address out of range error will always cause a processor exit. Therefore, any attempt to disable this type of exit (by specifying a value not included in this set) on a model 176 will cause an informative message to be issued to the dayfile.

Instruction stack purging on CYBER 180-class models can be made normal or extended by supplying an appropriate value for p. The initial status of stack purging is site-selectable. When stack purging is normal, the processor purges the instruction stack after executing an exchange jump, return jump, unconditional branch, or extended memory read instruction. When stack purging is extended, the processor also purges the instruction stack after central memory store and conditional branch instructions. Under normal circumstances, the CYBER 180-class models processor fetches the contents of the branch destination address prior to executing the branch; this could cause programs that do code modification to fail unless stack purging is extended.

An attempt to change the stack purging status on any machine other than CYBER 180-class models aborts the program.

Null or omitted parameters remain unchanged if fewer than three values are supplied on the macro call. Changes made to exit mode or stack purging status remain in effect for subsequent job steps unless explicitly changed by a MODE or MACHINE command or a MODE macro.

Examples:

Setting m to 6 on all models other than a model 176 selects error exiting on operand out of range and indefinite operand. On a model 176, address out of range cannot be deselected; it is automatically set by the system. Setting m to 6 or 7 on a model 176 would yield the same results.

Setting m to 13_8 on a model 176 selects underflow and error exiting on operand out of range. Setting m to 12_8 would also accomplish this.

SETASL (003)

The SETASL macro allows the user program to specify the account block SRU limit for a job. The account block limit is the maximum number of SRUs that can be accumulated by a job.

If s is greater than or equal to 77770_8, the account block SRU limit is set to the maximum for which you are validated (infinite if unlimited validation).

For interactive jobs, this function defines the number of SRUs allowed for a job before entry of another CHARGE command is required. For batch or interactive jobs, s represents the maximum SRU accumulation between two CHARGE commands or between one CHARGE command and the end of the job. The account block limit may not be set to a value less than the current job step limit.

Macro format:†

Location	Operation	Variable
	SETASL	s

s Account block SRU limit.

†This macro is not available in SYSTEXT. The user program must specify systems text PSSTEXT or call common deck COMCCMD (refer to appendix F).

SETJSL (003)

The SETJSL macro allows the user program to specify the job step SRU limit for each step of a job, including the current job step. The job step limit is the maximum number of SRUs that can be accumulated by a single job step. The job step limit must not exceed the account block limit (refer to SETASL macro).

Each user name and each charge/project number is validated for a maximum SRU limit. If s is greater than or equal to 77770_8, the job step SRU limit is set to the maximum for which you are validated (infinite if unlimited validation).

Macro format:†

Location	Operation	Variable
	SETJSL	s

s Job step SRU limit.

SETTL (003)

The SETTL macro changes the user program's CPU job step time limit for each job step, including the current job step. The CPU job step time limit is the amount of time (in seconds) that a job step is allowed to use the CPU.

Each user is validated for a maximum time limit (refer to the LIMITS command in Volume 3, System Commands). If t is equal to 77777_8, the time limit is set to the maximum for which you are validated (infinite if unlimited validation).

Macro format:†

Location	Operation	Variable
	SETTL	t

t Job step time limit ($1 \leq t \leq 77777_8$).

† This macro is not available in SYSTEXT. The user program must specify systems text PSSTEXT or call common deck COMCCMD (refer to appendix F).

EREXIT (004)†

NOS allows a user program to specify an error processing routine address. When an error occurs during program execution, the system causes the routine at that address to begin executing. If an additional error occurs while the error processing routine is executing, the job step aborts.

Macro format:

Location	Operation	Variable
	EREXIT	addr

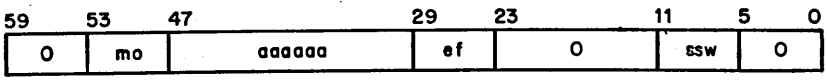
addr Address for error exit return.

As an example, a common procedure is for the program's preset routines to set the error exit processing.

```
                          .
                          .
                          .
Routine to      ERR       SA1     B0    READ ERROR RETURN INFO
complete                  .
processing                .
when an error             .
is encountered  PRS       ENDRUN        TERMINATE PROGRAM

                SUBR                    PROGRAM PRESET ROUTINE
                EREXIT    ERR           SET ERROR EXIT ADDRESS
                          .
                          .
                          .
                EQ        PRSX
                          .
                          .
                END
```

† It is recommended that you use extended reprieve processing (refer to section 10) rather than EREXIT wherever possible.

The following information is returned to location RA+0 if an error occurs at any time after an EREXIT request.



mo Mode of CPU error exit. For all computer systems except the model 176, the mode bit is set for the indicated EM register bit (refer to Exchange Package Area, appendix E).

mo Bit	EM Bit	Meaning
53	59	CM data error.†
52	58	CMC input error.†
51	57	EEM flag register operation parity error.†
50	50	Indefinite operand.
49	49	Operand out of range.
48	48	Address out of range.

If an mo bit is not set, the error is due to an illegal instruction.

For model 176, the following mode bits are set for the indicated PSD register bit (refer to Exchange Package Area, appendix E).

mo Bit	PSD Bit	Meaning
53	46	CM error condition.
52	--	Not used.
51	--	Not used.
50	38	Indefinite condition.
49	37	Overflow condition.
48	45	LCME block range condition.
48	44	CM block range condition.
48	43	LCME direct range condition.
48	42	CM direct range condition.
48	41	Program range condition.

The LCME error condition (PSD bit 47) and step condition (PSD bit 39) are not mapped to mo.

aaaaaa CPU address of the job when the error occurred.

ef Error flag. Refer to appendix E for a list of error flags.

ssw Status of sense switches.

NOTE

If REPRIEVE is used in the same routine as EREXIT, it will take precedence over EREXIT, and EREXIT will have no effect.

†Models 171, 172, 173, 174, 175, 720, 730, 750, 760, 865, and 875 only.

ROLLOUT (006)

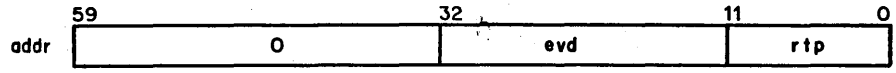
The ROLLOUT macro places a job in the rollout queue until a specified event occurs or for a fixed period of time. This function may be useful if a program requires a direct access permanent file that is currently busy.

Macro format:†

Location	Operation	Variable
	ROLLOUT	addr

addr — Address containing the time and/or event dependencies; if addr is not specified, the job is rolled out without time or event dependencies.

The following information must be set at location addr:



evd — Event descriptor. System programs use the EESET macro†† to make entries in a system event table indicating the occurrence of an event. The system compares the specified descriptor, evd, with events recorded in the table. If a match is detected, the job becomes eligible for rollin.

- evd≠0 — evd and rtp are placed in the control point area (TERW). When the job rolls out, the scheduler waits for the occurrence of evd or for the time period, rtp, to elapse before initiating rollin. Because the job may roll in for two different reasons, it is the user program's responsibility to verify whether the specified event actually occurred. Refer to system OPL common deck COMSEVT for a list of events.
- evd=0 — The job scheduler uses the value of evd already in the control point area. This option allows the user to roll out while waiting for a system-specified event. One such event is the event that is set up when an ATTACH request fails because of a busy file (refer to section 5). In this case, the event is the return of the file.
- evd=EXTM+00nn$_8$ (0200nn$_8$) — Specifies extended time rollout with no event dependency. The job rolls out for (7777$_8$ x nn) + rtp$_8$ job scheduler delay intervals. The symbol EXTM is defined in the system OPL common deck COMSEVT.

rtp — Rollout time period in job scheduler delay intervals (roughly 1 second each)††† where $0 \leq rtp \leq 4095$ (7777_8).

If rtp is zero, the scheduler uses the value of rtp already in the control point area. If evd is also zero, the job rolls out waiting for a system specified event. If the event goes undetected or never occurs, the job rolls in after a system-determined time interval.

---

†This macro is not available in SYSTEXT. The user program must specify systems text PSSTEXT or call common deck COMCMAC (refer to appendix F).

††Obtain a listing of COMCMAC for internal documentation of the EESET macro and the ROLLOUT macro and for a description of valid event descriptors.

†††The value of the job scheduler delay interval varies from site to site. Contact your site analyst to determine the value at your site.

SETSSM (010)

The SETSSM macro enables the user program to set or clear the secure system memory (SSM) flag. Setting this flag prevents the dumping of any portion of the job field length.

NOTE

While the SSM flag is set, certain system requests cannot be performed (refer to Security Considerations, section 1). In addition, certain system commands cannot be issued immediately following a program in which the SSM flag is set. Refer to Security Control in Volume 3, System Commands, for a list of these commands.

Macro format:†

Location	Operation	Variable
	SETSSM	p

p Clear SSM flag if lower 12 bits of p are zero; set SSM flag if lower 12 bits of p are nonzero.

† This macro is not available in SYSTEXT. The user program must specify systems text PSSTEXT or call common deck COMCMAC (refer to appendix F).

ONSW (011)

The ONSW macro sets the sense switches for a program (refer to figure E-1). This allows the user program to set switches for options for subsequent job steps in the job.

The sense switches reside both in the job's control point area and in bits 11 through 6 of RA+0 of the job field length. The field in RA+0 may be used by the user program to check sense switch settings; however, changing a bit in RA+0 will not cause the bit in the control point area to be changed. The field in RA+0 is set equal to the field in the control point area at the beginning of each job step, or if an error occurs. An ONSW (or OFFSW) request updates both fields separately.

The bit position specifies the switch to be set.

Macro format:

Location	Operation	Variable
	ONSW	n

n — Switches to be set; $0 \leq n \leq 77_8$; bit 0 corresponds to switch 1, bit 1 corresponds to switch 2, and so forth. If a bit is set, the corresponding switch is set; for example, ONSW 52_8 sets switches 2, 4, and 6. If $n=77_8$, all switches are set. If n=0, all switches remain unchanged.

OFFSW (012)

The OFFSW macro enables the user program to clear the sense switches for the job. Refer to the description of the ONSW macro for a discussion of sense switch settings.

Macro format:

Location	Operation	Variable
	OFFSW	n

n — Switches to be cleared; $0 \leq n \leq 77_8$; bit 0 corresponds to switch 1, bit 1 corresponds to switch 2, and so forth. If a bit is set, the corresponding switch is cleared; for example, OFFSW 52_8 clears sense switches 2, 4, and 6. If $n=77_8$, all switches area cleared. If n=0, all switches remain unchanged.

GETJN (013)

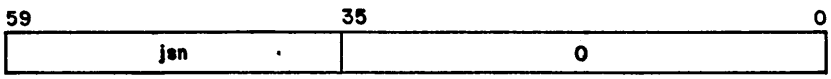
The GETJN function allows the user program to determine the job sequence name (JSN) of the job. For the format of the JSN, refer to Volume 3, System Commands.

Macro format:†

Location	Operation	Variable
	GETJN	addr

addr Address to receive job sequence name, left-justified.

The following information is returned to location addr:



jsn Job sequence name for the job.

GETPR (015)

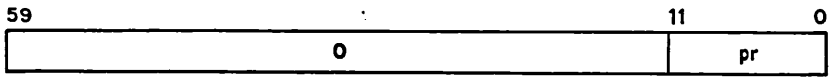
The GETPR function allows the user program to determine the CPU priority of the current job.

Macro format:†

Location	Operation	Variable
	GETPR	addr

addr Address where the CPU priority is returned.

The following information is returned to location addr:



pr CPU priority.

† This macro is not available in SYSTEXT. The user must specify systems text PSSTEXT or call common deck COMCMAC (refer to appendix F).

GETEM (016)

The GETEM macro enables the user program to determine under what exit mode the job is currently running.

Macro format:†

Location	Operation	Variable
	GETEM	addr

addr Address for return of current exit mode in bits 11 through 0.

GETASL (017)

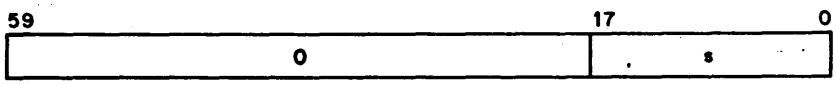
The GETASL macro returns the account block SRU limit for the job (refer to SETASL macro) to the specified address.

Macro format:†

Location	Operation	Variable
	GETASL	addr

addr Address to receive account block SRU limit.

The following information is returned to location addr:



s Account block SRU limit.

†This macro is not available in SYSTEXT. The user program must specify systems text PSSTEXT or call common deck COMCMAC or COMCCMD (refer to appendix F).

GETJSL (017)

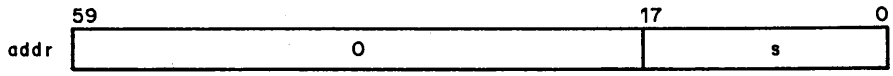
The GETJSL macro returns the job step SRU limit for the current job step (refer to SETJSL macro) to the specified address.

Macro format: †

Location	Operation	Variable
	GETJSL	addr

addr Address to receive job step SRU limit.

The following information is returned to location addr:



s Job step SRU limit.

GETTL (017)

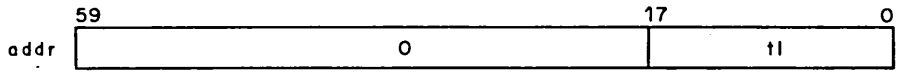
The GETTL macro returns the time limit for the current job step.

Macro format: †

Location	Operation	Variable
	GETTL	addr

addr Address to receive time limit.

The following information is returned to location addr:



tl CPU job step time limit (in seconds).

† This macro is not available in SYSTEXT. The user program must specify systems text PSSTEXT or call common deck COMCCMD or COMCMAC (refer to appendix F).

SETLC (022)

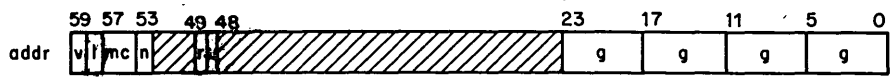
The SETLC macro enables the user program to set the loader control word for subsequent loader requests.

Macro format:†

Location	Operation	Variable
	SETLC	addr

addr Address of the loader control word to be set.

The following information must be set in location addr:



Field	Bit(s)	Description
v	59	Map validation bit. If bit is not set, default map option is used.
l	58	If bit is set, global library set is active.
mc	57-54	Map control bits (octal):
		00 No map is produced. 01 Statistics and errors. 02 Block assignments. 03 Partial map providing statistics, errors, and block assignments. 04 Entry points. 14 External references and entry points. 17 Full map providing information given individually by control bits 01, 02, 03, and 14.
n	53	Prevents field length reduction after relocatable load.
r	49	If bit is set, request relocatable load from user library.
u	48	If bit is set, program was loaded from a user library.
g	23-0	Global library set indicators.

The global library set indicators, contained in the loader control word (location LB1W in the user's control point area), are maintained by the SETLC macro. The user program should call the GETLC macro before using the SETLC macro to avoid destroying the library set information. With the GETLC macro, the user program can obtain the current loader control word. The program can then change the v, mc, and n fields, and issue a SETLC macro.

---

†This macro is not available to SYSTEXT. The user program must specify systems text PSSTEXT or call common deck COMCCMD (refer to appendix F).

SETRFL (023)

The SETRFL macro sets the initial central memory and extended memory field lengths for a job step. The specified values are used unless the system encounters one of the following:

- A system routine whose field lengths are specified during system creation.
- A routine that specifies the amount of CM or extended memory field length required in a 54 loader table (refer to CYBER Loader Reference Manual).
- An MFL or RFL command (refer to Volume 3, System Commands) or a subsequent SETMFL or SETRFL macro.

If you do not use the SETRFL function or the RFL command, the operating system determines how much field length to assign initially for each job step.

The CM field length is rounded upward to a multiple of 100_8 words. The extended memory field length is rounded upward to a multiple of 1000_8 words.

The values specified for CM field length and EM field length cannot exceed the current MFL values for the job (refer to the SETMFL macro in this section).

Macro format:†

Location	Operation	Variable
	SETRFL	n,m

n — New CM field length parameter.

m — New extended memory field length parameter.

NOTE

This request does not change the amount of memory currently available for use by the user program. To change the amount of memory currently available for use, use the MEMORY macro.

† This macro is not available in SYSTEXT. The user program must specify systems text PSSTEXT or call common deck COMCCMD (refer to appendix F).

GETJCR (024)†

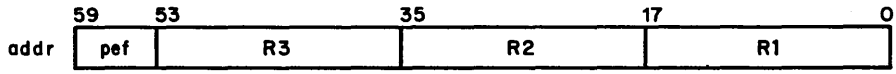
The GETJCR macro enables the user program to interrogate the job control registers associated with the job (refer to the description of execution control commands in Volume 3, System Commands). The last error flag encountered can also be determined.

Macro format: ††

Location	Operation	Variable
	GETJCR	addr

addr — Address for return of the job control registers.

The following information is returned to location addr:



pef — The last error flag encountered (refer to the EREXIT function for the values).

R3 — Job control register 3.

R2 — Job control register 2.

R1 — Job control register 1.

† You may choose to use the GETJCI macro instead of GETJCR, since GETJCI returns more information.

†† This macro is not available in SYSTEXT. The user program must specify systems text PSSTEXT or call common deck COMCMAC (refer to appendix F).

SETJCR (025)†

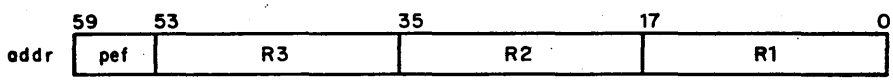
The SETJCR macro enables the user program to set the job control registers for the job. Refer to the description of the GETJCR macro for the format of the word.

Macro format:††

Location	Operation	Variable
	SETJCR	addr

addr — Address of the word containing the job control registers to be set.

The following information must be set at location addr:



pef — Previous error flag.

R3 — Job control register 3.

R2 — Job control register 2.

R1 — Job control register 1.

SETSS (026)

The SETSS macro changes the interactive subsystem under which your job is currently executing, provided you are validated for that subsystem.

Macro format:††

Location	Operation	Variable
	SETSS	ss

ss — Subsystem ordinal (refer to the TSTATUS macro, section 11, for a description of subsystem ordinals).

†You should use the SETJCI macro instead of SETJCR.
††This macro is not available in SYSTEXT. The user program must specify systems text PSSTEXT or call common deck COMCMAC (refer to appendix F).

GETJO (027)

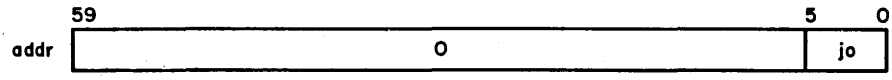
The GETJO macro returns the job origin code to the specified address.

Macro format:†

Location	Operation	Variable
	GETJO	addr

addr — Address to receive job origin code. Refer to appendix E for legal job origin codes.

The following information is returned to location addr:



jo — Job origin code.

GETJA (030)

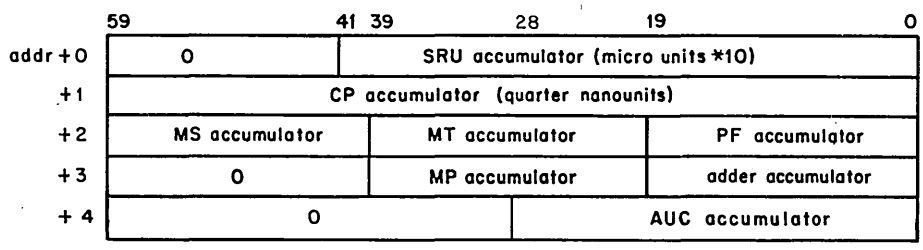
The GETJA macro returns a five-word reply block containing job accounting information beginning at the specified address.

Macro format:†

Location	Operation	Variable
	GETJA	addr

addr — Address of reply block.

The following information is returned to the five-word reply block beginning at location addr:



Nanounits are calculated as nanoseconds multiplied by the CPU accounting multiplier (refer to Resource Accounting in section 5 of the NOS 2 Administration Handbook).

†This macro is not available in SYSTEXT. The user program must specify systems text PSSTEXT or call common deck COMCMAC (refer to appendix F).

USECPU (031)

The USECPU macro specifies which central processor is to be used when more than one is available.

Macro format:

Location	Operation	Variable
	USECPU	n

n	Description
0	Either central processor can be used.
1	CPU 0 is to be used (the CPU with functional units on a 6700 or CDC CYBER 74-2x).
2	CPU 1 is to be used (the CPU without functional units on a 6700 or CDC CYBER 74-2x).

USERNUM (032)

The USERNUM macro returns the user name under which the job is running to the specified address.

Macro format:

Location	Operation	Variable
	USERNUM	addr

addr	Address to receive the user name, left-justified and zero-filled.

GETFLC (033, 055)

The GETFLC macro returns the field length control word from the job's control point area to the specified address.

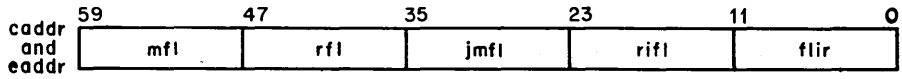
Macro format: †

Location	Operation	Variable
	GETFLC	caddr,eaddr

caddr Address to receive the central memory field length control word.

eaddr Address to receive the extended memory field length control word.

The GETFLC request can include either or both of the addresses. The field length control words are returned in the following format (multiply each 12-bit field by 100_8 for caddr or 1000_8 for eaddr to obtain the actual value).



mfl Maximum CM or extended memory field length for the current job step. This value may be reset with the MFL command (refer to Volume 3, System Commands) or the SETMFL macro.

rfl Initial running CM or extended memory field length for a job step. This value is always less than or equal to mfl and is set with the RFL command or the SETRFL macro. A value of zero indicates that the system controls the field length.

jmfl Maximum CM or extended memory field length for the entire job. The jmfl represents the upper bound on mfl.

rifl CM or extended memory field length for the job when it is to be rolled in.

flir CM or extended memory field length increase requested.

† This macro is not available in SYSTEXT. The user program must specify systems text PSSTEXT or call common deck COMCMAC (refer to appendix F).

PACKNAM (035)

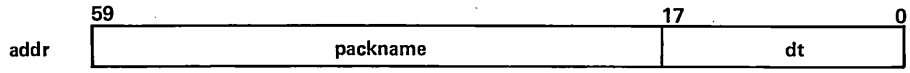
The PACKNAM (035) request specifies a default pack name and device type for your job. It allows the user program to omit the pn and r parameters from PFM requests for files residing on the auxiliary device. If a PACKNAM request is in effect and you wish to access permanent files on another auxiliary device, you can either specify the pn and r parameters in the PFM request or issue another PACKNAM macro. To access permanent files on your permanent file family device while a PACKNAM request is in effect, either specify a display code zero as the pn parameter in the PFM request or issue another PACKNAM macro with a binary zero pack name.

Macro format:†

Location	Operation	Variable
	PACKNAM	addr

addr Address of parameter word.

Parameter word format:



packname One- to seven-character alphanumeric pack name, left-justified, zero-filled. If this field is binary zero, any existing pack name and device type are cleared.

dt Three-character alphanumeric device type. The first two characters represent the type of auxiliary device (refer to appendix E); the third character indicates the number of spindles in a multispindle device. If the third character is omitted, the default is 1; if the entire field is zero, the default is the system pack type (set by the installation).

†This macro is not available in SYSTEXT. The user program must specify systems text PSSTEXT or call common deck COMCCMD (refer to appendix F).

PACKNAM (036)

The PACKNAM (036) request returns the default pack name and device type to the specified address.

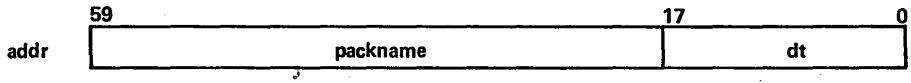
Macro format:†

Location	Operation	Variable
	PACKNAM	addr,N

addr — Address to which the pack name and device type are to be returned.

N — Indicates that the pack name and device type are to be returned.

The following information is returned to location addr:



packname — One- to seven-character alphanumeric default pack name, left-justified, zero-filled. (If no pack name is set, binary zero is returned.)

dt — Three-character alphanumeric default device type. The first two characters represent the type of auxiliary device (refer to appendix E); the third character indicates the number of spindles in a multispindle device.

GETSS (037)

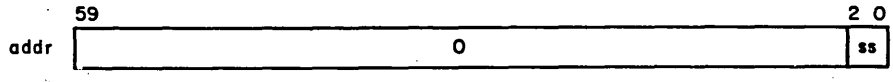
The GETSS macro returns the current subsystem ordinal to the specified address.

Macro format:††

Location	Operation	Variable
	GETSS	addr

addr — Address to receive subsystem ordinal (refer to TSTATUS macro, section 11, for a description of subsystem ordinals).

The following information is returned to location addr:



ss — Subsystem ordinal.

†This macro is not available in SYSTEXT. The user program must specify systems text PSSTEXT or call common deck COMCCMD (refer to appendix F).

††This macro is not available in SYSTEXT. The user must specify systems text PSSTEXT or call common deck COMCMAC (refer to appendix F).

VERSION (044)

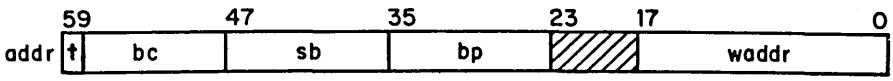
The VERSION macro returns the version name of the operating system (and, optionally, the system title line) from central memory to the specified location.

Macro format:†

Location	Operation	Variable
	VERSION	addr

addr Address of word containing macro parameters.

The following information must be specified at location addr:



t If set, indicates that the system title line is to be returned in addition to the operating system version name.

bc Number of bytes (1 to 10 or, if t is specified, 1 to 30) to return from source field (CM location containing version name and, optionally, system title).

sb Byte in source field at which to begin transfer (0 to 9 or, if t is specified, 0 to 29); the sum of bc and sb must be less than 11 (less than 31 if t is specified).

bp Byte position within receiving field (waddr) at which to begin transfer (0 to 4).

waddr Beginning address of three-word block (seven-word block if t is specified) to receive data.

† This macro is not available in SYSTEXT. The user program must specify systems text PSSTEXT or call common deck COMCCMD or COMCMAC (refer to appendix F).

GETLC (045)

The GETLC macro returns the loader control word to the specified address. Refer to the SETLC macro for the format of the loader control word.

Macro format: †

Location	Operation	Variable
	GETLC	addr

addr — Address to receive the loader control word.

GETGLS (046)

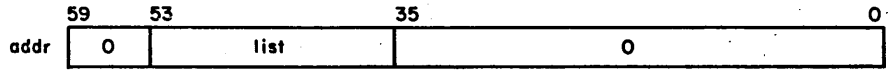
The GETGLS macro returns the global library set from the job's control point area. Refer to the CYBER Loader Version 1 Reference Manual for a discussion of global library sets. A parameter word is used to specify where a list of logical file names is to be placed.

Macro format: ††

Location	Operation	Variable
	GETGLS	addr

addr — Address of parameter word.

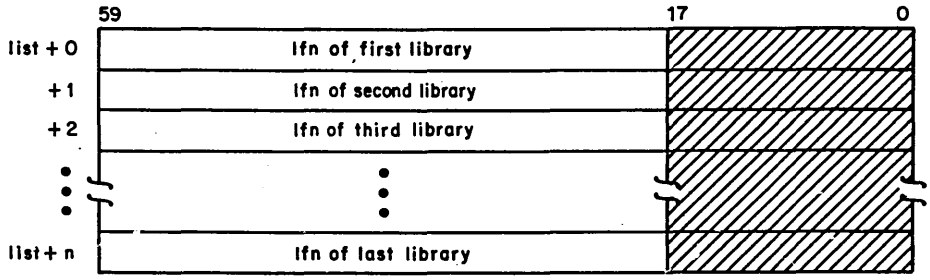
Parameter word format:



list — Beginning address to which the logical file names (left-justified) contained in the global library set are written. The value of this parameter is updated to the address of the word following the last library (list+n+1) upon completion.

†This macro is not available in SYSTEXT. The user program must specify systems text PSSTEXT or call common deck COMCCMD or COMCMAC (refer to appendix F).
††This macro is not available in SYSTEXT. The user program must specify systems text PSSTEST or call common deck COMCMAC (refer to appendix F).

The following information is returned to locations list through list+n:



n — The number of libraries defined in the global library set minus one.

lfn — The name of a global local library or the name of a system library.

SETGLS (047)

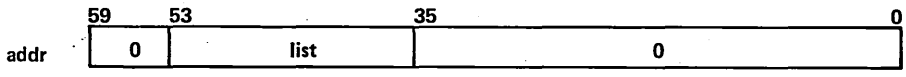
The SETGLS macro enables the user program to define the global library set indicators in the job's control point area. Refer to the CYBER Loader Version 1 Reference Manual for a discussion of global library sets. A parameter word specifies where the list of logical file names used to define the global library set is located.

Macro format:†

Location	Operation	Variable
	SETGLS	addr

addr — Address of parameter word.

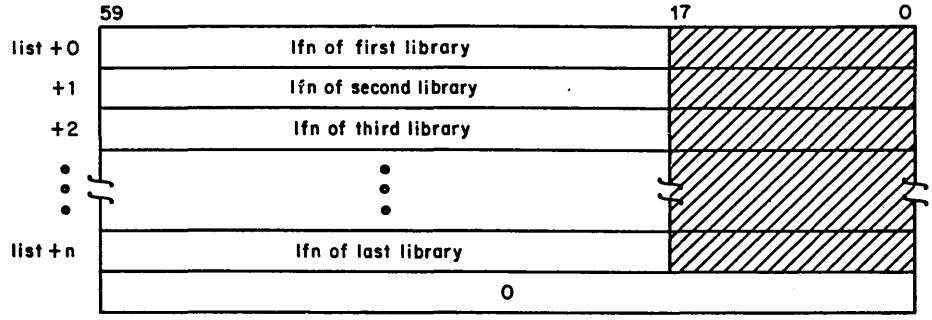
Parameter word format:



list — Beginning address to which the logical file names (left-justified) contained in the global library set are written. The value of this parameter is updated to the address of the last library (list+n) upon completion. If an error is encountered, list will be the address of the lfn of the library that caused the error.

† This macro is not available in SYSTEXT. The user program must specify systems text PSSTEXT or call common deck COMCMAC (refer to appendix F).

Before calling the SETGLS macro, locations list through list+n+1 must be as follows:



n The number of libraries in the global library set minus one.

MACHID (050)

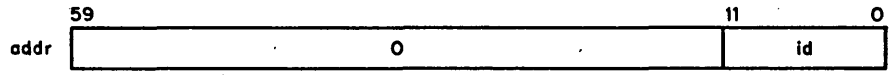
The MACHID macro enables the user program to determine the one- or two-character machine identification.

Macro format: †

Location	Operation	Variable
	MACHID	addr

addr Address to receive machine identification.

The following information is returned to location addr:



id Machine identification.

GETACT (051)

The GETACT macro obtains information regarding outstanding system activity for your job.

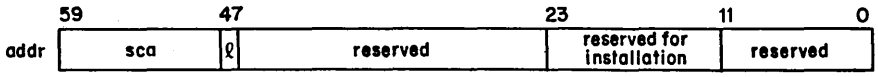
Macro format: †

Location	Operation	Variable
	GETACT	addr

addr Address to receive information.

† This macro is not available in SYSTEXT. The user program must specify systems text PSSTEXT or call common deck COMCCMD (refer to appendix F).

The following information is returned to location addr:



sca Count of system completable activities.

ℓ Long-term activity pending.

The sca field is incremented by one for each of the following:

- PP activity outstanding. This includes any PP routines called without auto recall that have not yet completed (for example, CIO). The following PPs are not included:
  - The PP that is processing the GETACT request.
  - PPs that have moved away from your job's control point at the time the GETACT request is being processed and then move back to your job's control point.
  - For jobs running under DIS control, the PP in which DIS is executing.
- Tape I/O activity outstanding or PP routine(s) in recall. Although the tape activity is initiated through CIO requests, it is handled separately by the system and is counted here, not with the PP activity.
- The rollout flag is set for your job.
- An interactive I/O request that has been made but not yet processed.
- Any SCP wait response indicators are set.

If ℓ (bit 47) is equal to 1, a long-term activity is pending. Such an activity requires either termination by the user or operator intervention. The following are long-term activities.

- K or L console display interface.
- CFO flag set (refer to Volume 3, System Commands, for a description of CFO entries).
- A system control point (SCP) long-term connection.
- DIS.

SETMFL (052)

The SETMFL macro enables you to change the job step maximum field length (MFL) boundary (refer to GETFLC and SETRFL macros for discussion of maximum field length limit). If either parameter is 0, the maximum field length is set to your maximum allowable field length.

Macro format:†

Location	Operation	Variable
	SETMFL	n,m

n — New maximum central memory field length limit for job step.

m — New maximum extended memory FL/1000_8 limit for job step.

GETPFP (057)

The GETPFP macro returns a job's permanent file parameters (current family name, pack name, device type, user name, and user index).

Macro format:†

Location	Operation	Variable
	GETPFP	addr

addr — Address to receive three-word table containing permanent file parameters.

The following information is returned to location addr:

	59	17	0
addr +0	family name		0
+1	pack name		device type
+2	user name		user index

†This macro is not available in SYSTEXT. The user program must specify systems text PSSTEXT or call common deck COMCCMD or COMCMAC (refer to appendix F).

GETLOF (061)

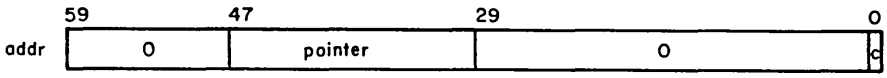
The GETLOF macro returns the list of files pointer for your job to the specified address. The list of files pointer is set with the SETLOF macro.

Macro format:†

Location	Operation	Variable
	GETLOF	addr

addr Address to receive the list of files pointer.

The following information is returned to location addr:



pointer Address of list of files table established with the SETLOF macro.

c Completion bit.

If the system returns 0 in the pointer field, no list of files address has been previously set (refer to description of SETLOF macro).

SETLOF (062)

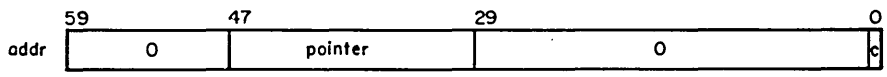
The SETLOF macro enables the user program to specify a pointer to a list of files whose circular buffers will be flushed at job step abort or, for terminal files, when the job is rolled out or when the user program issues a read with recall request on INPUT.

Macro format:†

Location	Operation	Variable
	SETLOF	addr

addr Address containing the pointer to the list of files table.

The following information must be specified at location addr:

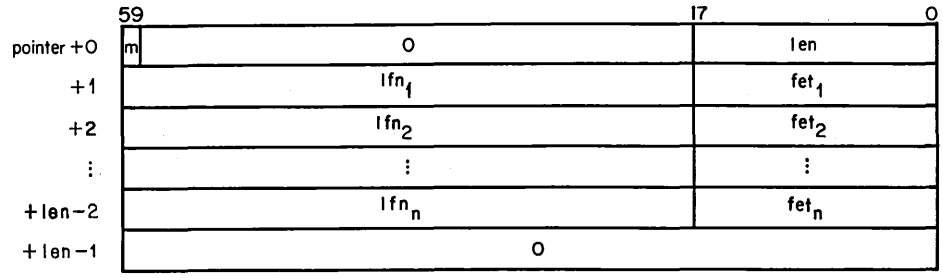


pointer Address of list of files table.

c Completion bit.

†This macro is not available in SYSTEXT. The user program must specify systems text PSSTEXT or call common deck COMCCMD (refer to appendix F).

The list of files table has the following format.



m	Common memory manager (CMM) block indicator; setting this bit indicates that the file list is CMM controlled.
len	Length of the list of files table including zero termination word.
lfn_i	File name.
fet_i	FET address.

The length of the list of files list must not exceed 7777_8 words.

When used with an interactive job step, the first file in the list must be the terminal output file.

During job step completion and after an error condition has occurred, the system uses the list of files to determine which files to complete for the user program. If the file name is OUTPUT or the FET has the flush bit set and the flushing criteria are met (refer to the description of the flush bit, FET Description in section 2), the data in the circular buffer is written to the specified file.

GETJCI (074)

The GETJCI macro returns job control information to the specified address.

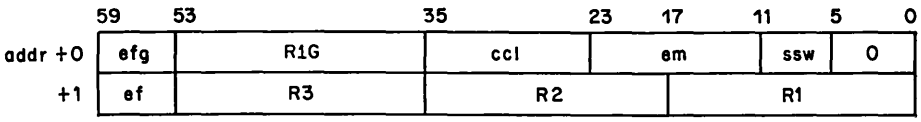
Macro format: †

Location	Operation	Variable
	GETJCI	addr

addr Address of table to receive job control information.

† This macro is not available in SYSTEXT. The user program must specify systems text PSSTEXT or call common deck COMCCMD (refer to appendix F).

The following information is returned to the two words beginning at location addr.



efg — Global error flag (refer to Volume 3, System Commands, for a description of global error flag).

R1G — Global register.

ccl — Reserved for system use.

em — Exit mode (as described in appendix E).

ssw — Sense switches (as described for ONSW macro).

ef — The last error flag encountered (as described for EREXIT macro).

R3 — Job control register 3.

R2 — Job control register 2.

R1 — Job control register 1.

SETJCI (074)

The SETJCI macro sets job control information with values obtained from a two-word table.

Macro format: †

Location	Operation	Variable
	SETJCI	addr

addr — Address of table specifying job control information.

The job control information must be established in the same format as that described for the GETJCI macro. The exit mode field is ignored for all jobs and the ccl field is ignored for user jobs.

† This macro is not available in SYSTEXT. The user program must specify systems text PSSTEXT or call common deck COMCCMD (refer to appendix F).

PROTECT (075)

The PROTECT macro enables the user program to select or deselect the following job control features.

- Extended memory field length preservation over job steps.
- User file privacy.

Macro format:†

Location	Operation	Variable
	PROTECT	em,ufp

em — ON to select extended memory preservation; OFF to deselect extended memory preservation.

ufp — ON to select user file privacy; OFF to deselect user file privacy.

To specify extended memory field length preservation, you must be authorized to use the PROTECT command (refer to LIMITS command in Volume 3, System Commands).

If the user program issues the PROTECT macro with the ufp parameter set to ON, all files that are subsequently created or accessed in the current job step are returned at job step termination. If a PROTECT macro with the ufp parameter set to OFF is subsequently issued in the same job step, only those files created or accessed prior to the second PROTECT macro are returned.

If the user program selects user file privacy, a checkpoint of the job step in which the PROTECT function is issued is not allowed.

SETJOB (106)

The SETJOB macro allows the user program to specify or to change the user job name (UJN), the disposition code for end-of-job output, and the end-of-job option. The user program must create a two-word parameter block containing this information before issuing the SETJOB call.

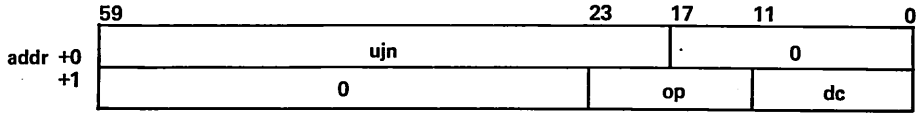
Macro format:†

Location	Operation	Variable
	SETJOB	addr

addr — Address of parameter block.

†This macro is not available in SYSTEXT. The user program must specify systems text PSSTEXT or call common deck COMCCMD (refer to appendix F).

Parameter block format:



ujn — One- to seven-character alphanumeric UJN for the current job (left-justified, zero-filled). If binary zero, the current UJN remains in effect. The system initially assigns each job a UJN equal to the four-character user index hash or, for batch jobs, the name specified by the Job command.

op — End-of-job option.

op	Description
SU	Suspend job.
TJ	Terminate job.

dc — Disposition of output at end of job.

dc	Description
DF	Default queue for job; depends on job origin type (refer to appendix E).
NO	No output.
TO	Wait queue.

If a field is set to binary zero, the corresponding option is not changed.

GETSSID (110)

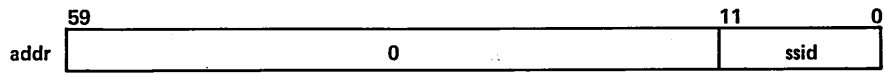
The GETSSID macro returns the subsystem identifier to the specified address.

Macro format:†

Location	Operation	Variable
	GETSSID	addr

addr — Address to which subsystem identifier is returned.

The following information is returned to location addr:



ssid — Subsystem identifier.

†This macro is not available in SYSTEXT. The user program must specify systems text PSSTEXT or call common deck COMCMAC (refer to appendix F).

GETJOSC (111)

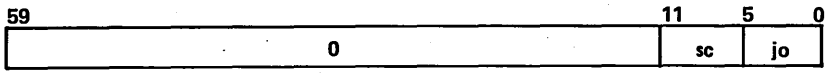
The GETJOSC macro returns the job origin and service class values to the specified address.

Macro format:†

Location	Operation	Variable
	GETJOSC	addr

addr — Address to which job origin and service class values are returned. Refer to appendix E for a list of job origin and service class values.

The following information is returned to location addr:



sc — Service class.

jo — Job origin.

SHELL (113)

The SHELL macro allows you to specify controls for a shell program (refer to Volume 3, System Commands, for a description of shell programs).

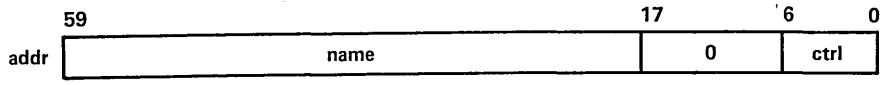
Macro format:†

Location	Operation	Variable
	SHELL	addr

addr — Address containing control information for shell program.

†This macro is not available in SYSTEXT. The user program must specify systems text PSSTEXT or call common deck COMCMAC (refer to appendix F).

The control information at address addr must have the following format.



name One- to seven-character name of shell program.

ctrl Control information for shell program.

Bit	Description
7	Clear control information if there are SHELL load errors. (If this bit is not set, the job is terminated on a SHELL load error.)
6	Clear control information if shell program aborts. This bit must be set if a project epilogue is defined for the current account block.
5	Allow load of shell program from local file.†
4	Allow load of shell program from global library.
3	Allow load of shell program from system library.
2	Call shell program during interactive job when there are no more commands to process.
1	Call shell program for each command outside a procedure.
0	Call shell program for each command inside a procedure.

Example:

To allow shell program ABC to be loaded from a global library and to monitor all commands, a user program could include the following statements.

```
          .
          .
          .
        SHELL   CTRL1
          .
          .
          .
CTRL    SET     1S4+1S2+1S1+1S0
CTRL1   VFD     42/3LABC,11/0,7/CTRL
          .
          .
          .
```

Appendix L contains an illustration of a shell program that permits the use of non-NOS syntax commands.

†A shell program loaded from a local file must be in ABS or OVL format. You must ensure that there is sufficient field length to load the program.

GETJAL (116)

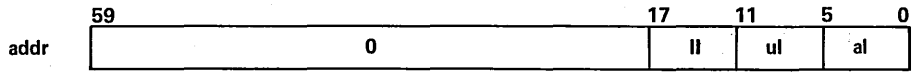
The GETJAL macro returns your job's current security access level and the job's security access level limits to the specified address.

Macro format:†

Location	Operation	Variable
	GETJAL	addr

addr — Address to which the job's security access level information is to be returned.

The following information is returned to location addr:



ll — The numerical value of the job's lower access level limit.
ul — The numerical value of the job's upper access level limit.
al — The numerical value of the job's current access level.

The system OPL common deck COMCVLC can be used to convert access level names to numerical values and vice versa.

On an unsecured system, GETJAL returns a zero word to location addr.

SETJAL (117)

The SETJAL macro changes the security access level of your job. This macro is valid only on a secured system.

Macro format: †

Location	Operation	Variable
	SETJAL	addr

addr — Address of a word containing the numerical value of the new access level for the job, right-justified with binary zero fill. The access level specified must be valid for your job (refer to Security Access Levels in section 1). The access level specified may not be lower than the current job access level unless you have special authorization.

†This macro is not available in SYSTEXT. The user program must specify systems text PSSTEXT or call common deck COMCMAC.

GETUSC (123)

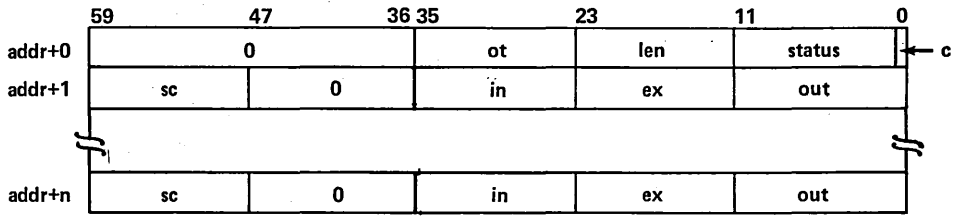
The GETUSC macro returns the service classes valid for the user program for a particular job origin type and indicates the upper and lower priority bounds for each of those service classes. The user program must contain a parameter block consisting of one word of parameters for the request and one word for each service class to be returned.

Macro format:†

Location	Operation	Variable
	GETUSC	addr

addr Address of parameter block.

Parameter block format:



The user program enters the following information into the parameter block:

Field	Location	Description
ot	35-24	The job origin type for which service class information is to be returned.
len	23-12	The maximum number of service classes for which information is to be returned. The symbol MXJC, defined in common deck COMSSCD, can be used to specify the number of service classes in the system. Service classes are returned in increasing numerical order (refer to appendix E for a list of valid service classes and the numerical value associated with each class).
c	0	Completion bit (must be set to zero).

†This macro is not available in SYSTEXT. The user program must specify systems text PSSTEXT or call common deck COMCMAC (refer to appendix F).

The user program must enter zeros in the remainder of word addr+0.

The following information is returned to the parameter block:

Field	Location	Description
len	23-12	The actual number of service classes returned.
status	11-1	Error status.

Value	Description
0	No error.
1	Value of zero specified for len.
2	Undefined origin type.
3	Completion bit set in request.

c	0	Completion bit (set when processing is complete).
sc	59-48	Two-character display code mnemonic for the service class.
in	35-24	Lower bound for input files in this service class.
ex	23-12	Priority upper bound; the highest possible priority for executing jobs in this service class.
out	11-0	Priority lower bound; the lowest possible priority for output files in this service class.

In the following example, the GETUSC macro is used to return information for all service classes for interactive origin type.

```
          .
          .
          .
OPL       XTEXT   COMCMAC            REQUIRED FOR GETUSC MACRO
OPL       XTEXT   COMSSCD            DEFINES MXJC
REPLY     VFD     24/0,12/IAOT,12/MXJC,12/0
          BSSZ    MXJC
           .
           .
           .
          GETUSC  REPLY
```

SETSC (124)

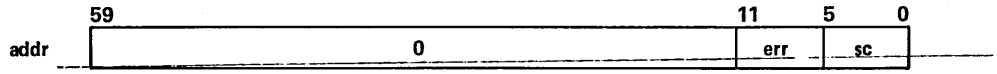
The SETSC macro changes the service class of your job.

Macro format:†

Location	Operation	Variable
	SETSC	addr

addr Address of parameter word.

Parameter word format:



The user program enters the following information into the parameter word:

sc Mnemonic or numerical value for the service class to which the job is to be assigned (refer to appendix E for a list of valid service classes and the mnemonic and numerical value associated with each class); you must use the system OPL common deck COMSSCD in order to specify a service class mnemonic.

CPM returns the following information to the parameter word:

err Error status.

Value	Description
0	No error.
1	Undefined service class.
2	Service class not valid for user and job origin.
3	Service class full.

In the following example, the job's service class is changed to batch service class (BCSC).

```
          .
          .
          .
OPL       XTEXT   COMSSCD
OPL       XTEXT   COMCMAC
PARAM     VFD     54/0,6/BCSC
          .
          .
          .
          SETSC   PARAM
```

†This macro is not available in SYSTEXT. The user program must specify systems text PSSTEXT or call common deck COMCMAC (refer to appendix F).

GETPAGE (127)

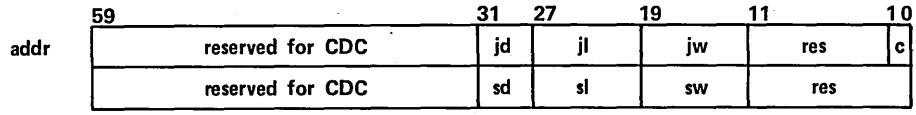
The GETPAGE macro returns page size parameters to the specified address.

Macro format:†

Location	Operation	Variable
	GETPAGE	addr

addr Address to which page size parameters are returned.

The following information is returned to location addr:



jd Job print density in lines per inch (6 or 8).
jl Job page length in lines.
jw Job page width in characters.
c Complete bit.
sd System print density in lines per inch.
sl System page length in lines.
sw System page width in characters.

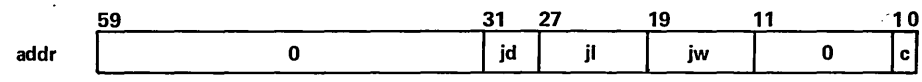
SETPAGE (130)

The SETPAGE macro allows the user program to specify page size parameters to control printed output.

Macro format:†

Location	Operation	Variable
	SETPAGE	addr

addr Address of the word containing page size parameters. All parameters must be specified with unused fields set to binary zero.



jd Job print density in lines per inch (6 or 8).
jl Job page length in lines ($16 \leq jl \leq 255$).
jw Job page width in characters ($40 \leq jw \leq 255$).
c Completion bit (must be zero on entry).

†This macro is not available in SYSTEXT. The user program must specify systems text PSSTEXT or call common deck COMCMAC.

GETLVL (133)

The GETLVL macro returns the PSR level of NOS that you are running.

Macro format:†

Location	Operation	Variable
	GETLVL	addr

addr — Address of the word to which the system level is to be returned. The system level is a binary number, right-justified with binary zero fill.

GETRI (134)

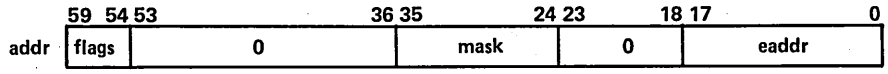
The GETRI macro returns information on your job's current reprieve status to the specified address.

Macro format:

Location	Operation	Variable
	GETRI	addr

addr — Address to which the reprieve information is to be returned.

The following information is returned to location addr:



flags — If bit 59 is set, EREXIT processing is enabled.
If bit 58 is set, normal reprieve processing is enabled.
If bit 57 is set, extended reprieve processing is enabled.
If bit 56 is set, normal reprieve is in progress.
If bit 55 is set, extended reprieve is in progress.

mask — Normal reprieve mask bits (if normal reprieve condition is set). Zero (otherwise).

eaddr — Extended reprieve parameter block address (if extended reprieve processing is enabled).

Error exit return address (if normal reprieve processing is enabled or if EREXIT processing is enabled). Zero (otherwise).

†This macro is not available in SYSTEXT. The user program must specify systems text PSSTEXT or call common deck COMCCMD (refer to appendix F).

GETCN (136)

The GETCN macro returns the charge and project number.

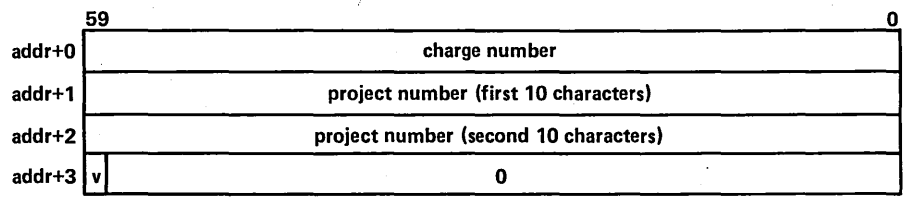
The GETCN macro returns a four-word block containing the current charge and project number combination beginning at the specified address. Word four, bit 59, will be clear if this is a nonvalidated charge and project number. Word four, bit 59, will be set if these are validated numbers. (The only case in which the numbers would not have been validated is if your validation did not require you to have a valid charge and project number and your job had never issued an explicit CHARGE command.)

Macro format:

Location	Operation	Variable
	GETCN	addr

addr Address of the reply block.

The following information is returned to the four-word reply block beginning at location addr:



v Validation bit.

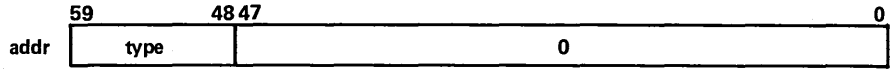
CSTATUS (137)

The CSTATUS macro returns the type of the system console.

Macro format:†

Location	Operation	Variable
	CSTATUS	addr

addr Address of the word to which the console type is to be returned.



type Console type of primary console:

0 = CC545
1 = CDC 721

†This macro is not available in SYSTEXT. The user program must specify system text PSSTEXT or call common deck COMCDCM.

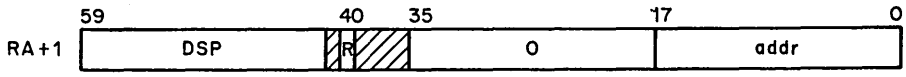
DSP FUNCTION

The dispose processor (DSP) performs file routing operations for NOS. DSP places a file in the specified queue at either the central site or a remote batch site. A user program can call DSP by issuing an RA+1 request or a ROUTE macro. A DSP call can be made from a job of any job origin type.

NOTE

If a file is routed to an input queue, it must contain a valid USER command or the job step initiating the route will be aborted with a security conflict. Refer to section 1 of this manual for more information on security conflict processing.

The format of the call to DSP is:



R Auto recall bit.

addr First word address of parameter block.

The user program must define a parameter block containing the following information before issuing a DSP RA+1 call or a ROUTE macro.

NOTE

Various symbols related to DSP calls and the DSP parameter block are defined in the system OPL common deck COMSDSP.

	59	47	41	35	23	17	11	0
addr +0	lfn					ec	reserved	c
+1	reserved	forms		disp	ex s	ic	flags	
+2	slid		dlid		tid			
+3	ujn					reserved	priority	
+4	spacing	service class		abort code	reserved	rc	reserved	
+5	reserved							
+6	reserved							

By setting bit 6 of the flags field in addr+1, the user program can define an extended DSP parameter block containing the following information.

	59	47	23	17	0
addr +7	dd	reserved for CDC	0	eflags	
+10	reserved for CDC			reserved for CDC	
+11	reserved for CDC			ertaddr	
+12	reserved for CDC				
+13	reserved for CDC				
+14	reserved for CDC				
+15	reserved for CDC				
+16	reserved for installations				

NOTE

In general, a field in the DSP parameter block (or extended block) will only be processed if the corresponding bit in the flags (or eflags) field is set.

lfn — Local file name of file to be routed (must be a queued file or a local file; must not reside on a removable device; and must not be an execute-only file). After the request has been completed, this contains the JSN of the queued file (if bit 17 of flags was set).

ec — Error code returned by system when bit 12 of flags field is set (error codes described later in this section).

c — Completion bit (must be zero when function is issued; system sets bit to one when operation complete).

forms Forms code or input flags. Two display code characters or specific bits set identifying the forms to be used for this file or other processing options. If file is routed to input queue, this field is defined as follows:

Bit(s)	Description
47-45	Reserved.
44	Send file to output queue with forced abort error code.
43	Send file to input queue even if Job command error occurs.
42-36	Reserved.

Forms codes consist of two alphanumeric characters and are assigned by each installation. You should contact installation personnel to determine what forms codes are available (if any).

disp Disposition code. Two alphanumeric characters specifying the disposition of the routed file.

Code	Description
IN†	Release file to input queue.
LP	Print on any line printer.
LR	Print on 580-12 line printer.
LS	Print on 580-16 line printer.
LT	Print on 580-20 line printer.
NO	Release file to input queue. Drop job output files at job completion.
PB	Punch system binary.
PH	Punch coded.
PL	Plotter.
PR	Same as LP.
PU	Same as PH.
P8	Punch 80 column binary.
SB	Same as PB.
SC	Rescind prior routing and change file type to local (LOFT).
TO	Input queue, output to wait queue.
TT	Wait.
WT	Same as TT.

ex External characteristics of print files:

ex	Description
0	Any external characteristics.
1	Not used.
2	ASCII graphic 48-character set (not supported; provided for NOS/BE compatibility).
3	CDC graphic 48-character set (not supported; provided for NOS/BE compatibility).
4	CDC graphic 63/64-character set.
5	ASCII graphic 63/64-character set.
6	ASCII graphic 95-character set.
7	Reserved.

†When a job that has been routed to IN runs to completion, the following local files (if they exist) will be routed to the appropriate queues: OUTPUT,PUNCH,PUNCHB,P8. You can prevent this implicit file routing with the SETJOB command or macro.

External characteristics of punch files:

ex	Description
0	System default.
1	System binary.
2	80-column binary.
3	Not used.
4	026 mode.
5	029 mode.
6	ASCII.
7	Reserved.

External characteristics of plot files:

ex	Description
0	Reserved.
1	Transparent 6-bit plotter data.
2	Transparent 8-bit plotter data.
3-6	Not used.
7	Reserved.

s Forced service class flag. If set, the specified service class will be used.

ic Internal characteristics (bits 19 through 18) code:

Value	Description
0	Display code.
1	ASCII code.
2	Binary.
3	Reserved.

If ic is not specified, the system assigns a default value based on the value of ex.

ex for Print Files	Default ic
2	1
3	0
4	0
5	0
6	1

ex for Punch Files	Default ic
1	2
2	2
4	0
5	0
6	1

flags Each bit set indicates that a particular parameter is specified.

Bit	Description
17	Job sequence name assigned to queued file by system is returned to addr+0, bits 59 through 18.
16	Reserved.
15	580 PFC spacing code.
14	Repeat count.
13	UJN is specified in addr+3, bits 59 through 18.
12	No dayfile message and return error code to addr+0, bits 17 through 12.
11	Reserved.
10	Forms code.
9	Priority.
8	Internal characteristics.
7	External characteristics.
6	Extended parameter block.
5	Reserved for installations.
4	Disposition code.
3	SLID/DLID.
2	TID.
1	Route to central site.
0	End-of-job (deferred ROUTE).

The system determines the routing destination of an output file by flag bits 1 and 2 (route to central site and tid, respectively) as follows:

Central Site Bit	tid Bit	Routing Activity
0	0	If the calling job is of remote batch origin, the system routes output to the remote batch queue and associates with it the user name and family of the originating remote batch terminal. If the calling job is any other origin type, the system routes output to the central site queue. If the file being routed was previously routed with queuing deferred, the previously specified destination remains in effect.

slid Three-character alphanumeric logical identifier (LID) of the source mainframe for the file. The following special values may also be entered in this field.

Value (Octal)	Description
0	Get slid from job input file system sector.
1	Set slid to zero.
2	Set slid to physical identifier of this mainframe.

dlid — Three-character alphanumeric logical identifier (LID) of the destination mainframe for the file. The following special values may also be entered in this field.

Value (Octal)	Description
0	Set dlid to zero.
1	Set dlid to zero.
2	Set dlid to physical identifier of this mainframe.

tid — For routing to a remote batch queue, this field should contain the complement of the address of a two-word block. The first word of the block contains the family name and the second word contains the user name (both left-justified and zero-filled). This is the user name that must be used to log in at a remote terminal to get the routed file.

If the address specified by the tid field equals 77777777_8, the file is routed to the remote batch queue with the family and user name of the calling job.

For routing to the local batch queue, the tid contains an ID code (right-justified).

ujn — The user job name (UJN) to be used instead of the job's default UJN.

Central Site Bit	tid Bit	Routing Activity
0	1	The system routes output according to the contents of the tid field (description follows).
1	0	The system routes output to the central site queue.
1	1	The system routes output to the central site queue if the tid field contains a valid ID code; otherwise, the system aborts the job.

priority — If job priority is greater than 7760 octal, the specified priority is used for output files (otherwise field is ignored).

spacing — Spacing code for output files (580 PFC). Octal value from 0 to 77.

service class — Two-character forced service class code in display code, specifying the service class of the routed file. Refer to appendix E for a list of valid service classes and service class codes.

abort code — If bit 44 of the DSP parameter word 1 is set, the value of this field determines which message from the following table is sent to the dayfile.

Value	Message
0001	Job command error.
0002	Card reader error.
0003	Operator input termination.
0004	Disk full.
0005	Disk parity error.

rc — Repeat count.

dd — The data declaration in display code; defines the data type of a file destined to a remote mainframe.

dd	Data Type
C6	6-bit character data.
C8	8-bit character data.
US	Undefined data type, structured file.
UU	Undefined data type, unstructured file.

eflags — Flags indicating which parameters are specified in the extended parameter block.

Bit	Description
17-1	Reserved.
0	Data declaration.

ertaddr — The first word address of the block containing the explicit remote text string for the file. The first word of the block contains the length of the string in characters; the rest of the block contains up to 300_8 display-code characters.

ROUTE

The format of the ROUTE macro is as follows:

Location	Operation	Variable
	ROUTE	addr,R

addr — Address of parameter block.

R — Auto recall bit.

Example 1:

The following example routes a file named PRINTF for printing on any central site line printer.

```
          .
          .
PBLK      VFD     42/0LPRINTF,18/0
          VFD     24/0,12/2HLP,6/0,18/22B
          VFD     60/0
          VFD     60/0
          VFD     60/0
          VFD     60/0
          VFD     60/0
          .
          .
          ROUTE   PBLK,R
          .
          .
```

Example 2:

The following example performs a deferred route of a file named PRINTF in ASCII graphic 95-character set and specifies that two copies be printed.

```
          .
          .
PBLK      VFD     42/0LPRINTF,18/0
          VFD     24/0,12/2HLP,3/6,3/0,18/40221B
          VFD     60/0
          VFD     60/0
          VFD     42/0,6/1,12/
          VFD     60/0
          VFD     60/0
          .
          .
          ROUTE   PBLK,R
          .
          .
```

Example 3:

The following example routes a file named JOB1 to the input queue for processing at the central site and writes the job name assigned by the system to the user dayfile.

```
          .
          .
PBLK      VFD     42/0LJOB1,18/0
          VFD     12/0,12/1000B,12/2HIN,6/0,18/402022B
          VFD     60/0
          VFD     60/0
          VFD     60/0
          VFD     60/0
          VFD     60/0
          .
          .
          ROUTE   PBLK,R
          SA1     PBLK
          MX0     42
          BX6     X0*X1
          SA6     A1
          MESSAGE PBLK,3
          .
          .
```

ERROR PROCESSING

When an error occurs in processing a ROUTE macro or DSP call, either a dayfile message explaining the error is issued, or an error code is returned in bits 17 through 12 of addr+0. If the address of the parameter block is outside the field length of the job or if the completion bit is set when the function is issued, the job step aborts. For all other errors, the function is not executed, but error processing continues. If bit 12 of the flags field is set, an error code is returned and no dayfile message is issued. If bit 12 is not set, no error code is returned, a dayfile message is issued, and the job step aborts.

The error codes that can be returned are as follows:

Error Codes	Description
1	File name error.
2	File not on mass storage.
3	Invalid file type.
4	Reserved.
5	Route to input not immediate.
6	Immediate routing, no file.
7	Incorrect disposition code.
10	User access not valid.
11	Undefined service class.
12	Incorrect request (unconditional abort).
13	Deferred routing not allowed.
14	Incorrect data declaration.
15	Incorrect LID.
16	Cannot route job input file.
17	Completion bit already set (unconditional abort).
20	File on removable device.
21	Incorrect TID.
22	Forms code not alphanumeric.
23	Incorrect internal characteristics.
24	Queued file read error.
25	QFT full.
26	This routing not allowed.
27	Device full.
30	Mass storage error.
31	I/O sequence error (unconditional abort).
32	Job command error.
33	Too many deferred batch jobs.
34	Incorrect USER command.
35	Device unavailable.
36	Incorrect file mode.
37	Incorrect external characteristics.
40	Incorrect origin type.
41	Incorrect spacing code.
42	Incorrect job abort code.
43	Incorrect owner user and family name.
44	Incorrect creation user and family name.
45	Service class incorrect for user or job origin.

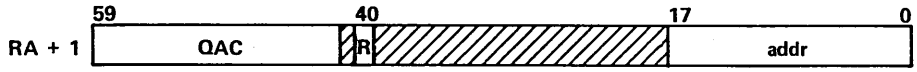
For a complete listing of error messages, refer to appendix B.

QUEUE ACCESS INTERFACE 8

The queue access (QAC) interface allows a user program to interact with the executing job table (EJT) and the queued file table (QFT). The user program can specify either a single entry (file or job) in a QAC request, or a group of entries. Requests are provided that allow a user program to:

- Attach files that are in the QFT (GET request).
- Determine the status of files in the QFT and jobs in the EJT (PEEK request).
- Alter or purge files in the QFT and jobs in the EJT (ALTER request).

A user program accesses the QAC interface by issuing an RA+1 call. The format of the call is:



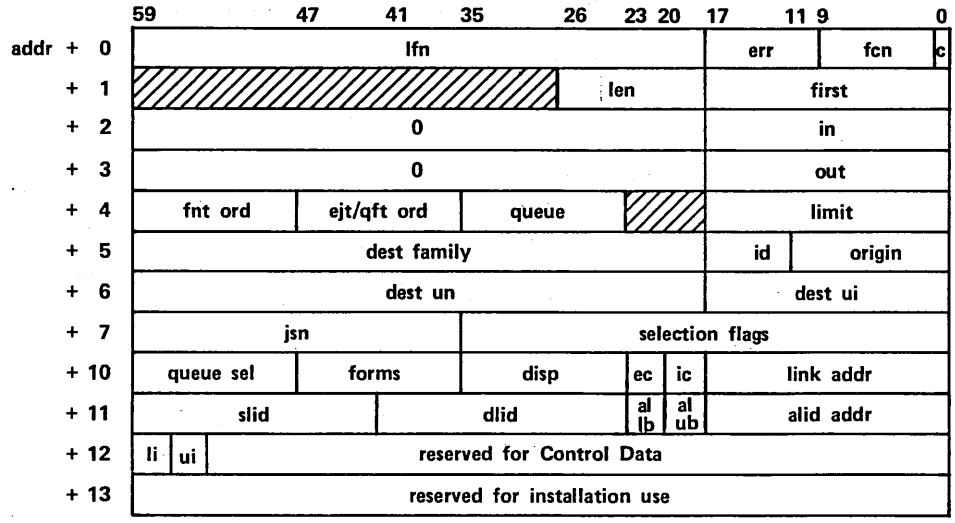
R Auto recall bit.

addr Address of the QAC parameter block; must be a valid address within your job's field length.

The user program must define a parameter block before issuing the RA+1 call. The first 14_8 words of this block are common to all requests.

NOTE

Various symbols related to QAC calls and the QAC parameter block are defined in system OPL common deck COMSQAC.



Field	Description
lfn	The user program sets this field for a GET request to the local file name to be given to the queued file when it is attached from the queue. If this field is set to zero, QAC assigns the job sequence name (JSN) of the file as the local file name.
err	QAC enters one of the following error codes upon completion of a request. Error codes 1 through 12_8 indicate nonfatal errors. During linked request processing, QAC continues with the next request when a nonfatal error is detected.

err	Description
0	The request completed without error.
1	Incorrect queue type. Indicates one of the following conditions. • Undefined queue bits (addr+10_8) were set. • For a GET request, the executing job queue was selected. • Bit 0, addr+7 is set, but more than one queue was selected in addr+4. • The user program entered an invalid queue type in the queue field of addr+4 during an interrupted PEEK request.
2	Incorrect family/user name. The specified family (addr+5) or user name (addr+6) is not active or failed validation.
3	Undefined origin type (addr+5). The specified origin type is not valid.
4	The user program sets one of the following combinations of bits. • Bits 3 and 4 of the selection flags field (addr+7). • Bits 9 and 10 of the selection flags field (addr+7). • Bits 0 and 1 of the alter flags field (addr+14_8). • Bits 10 and 11 of the alter flags field (addr+14_8). • Bits 12 and 16 of the alter flags field (addr+14_8). • Bits 9, 13, 14, and 15 of the alter flags field (addr+14_8).
5	Incorrect forms code. The forms code field (addr+17_8) for an ALTER request does not consist of two alphanumeric characters or a zero.

Field	Description
err	Description
6	Incorrect disposition code (addr+10_8, or addr+17_8 for an ALTER request), or the PU disposition code was selected but no external characteristics were specified to select a particular punch type.
7	File or job not found.
10_8	Duplicate local file name. For a GET request, QAC refused to attach a file because the file has the same name as an existing local file. QAC enters the name of the file that caused the error in the lfn field (addr+0). This condition does not result in an error if the inhibit duplicate lfn search flag (bit 1, addr+7) is set.
11_8	Incorrect batch ID. The batch ID specified (addr+5, or addr+16_8 for an ALTER request) is not less than 70_8.
12_8	Incorrect file name. The local file name (addr+0) specified for a GET request is not valid.
13_8	Count of zero incorrect. The maximum number of files or jobs to be selected in a PEEK request (addr+14_8 and addr+15_8) for at least one queue must be greater than zero.
14_8	The user program did not set any alter flags (addr+14_8) for an ALTER request, or the QAC buffer is not large enough to hold one PEEK reply for a PEEK request.
22_8	Incorrect external characteristics. The external characteristics code (addr+17_8) for an ALTER request is not valid.
23_8	The user program attempted to drop an interactive job.
24_8	Incorrect destination LID. The destination LID (addr+15_8) for an ALTER request is not valid.
25_8	Incorrect access level. Either the access level selection limits (addr+11_8) or the new access level for an ALTER request (addr+17_8) are not valid.
26_8	Incorrect file size index. The file size index selection limits (addr+12_8) are not valid.
27_8	Incorrect service class. The two-character service class was not valid for the user, or not valid for the current origin type of the specified job.
30_8	Service class full. The service class change cannot be made because the number of jobs with that class is already at the service limit.
31_8	Cannot change class of on-line job. The service class of another on-line job cannot be changed.

Field	Description

err	Description
32_8	Undefined service class. The service class mnemonic is not defined.
33_8	Job already waiting on service class. The service class change cannot be made because the job is waiting for a CLASS command in the job to complete.
34_8	Cannot change class of subsystem. The service class of a job that is executing at the subsystem service class cannot be altered.

fcn — The user program enters the function code number for the type of request.

Request	fcn
ALTER	1
GET	2
PEEK	3

c — Completion bit. The user program must set this bit to zero. QAC sets this bit to one when the request is complete.

len — The user program enters the total length of the QAC parameter block minus 5. The value of len for each type of request is as follows:

Request	len
ALTER	14_8
Extended GET	116_8
GET	13_8
PEEK	13_8

first — The user program enters the first word address of the buffer to be used by QAC. For an ALTER request, the user program can enter messages in the buffer; for a PEEK request, QAC enters the PEEK reply block in the buffer.

in — This field is a pointer to the next available word in the buffer to which information can be written. For a PEEK request, the user program must set this field to the same value as first.

out — This field is a pointer to the next word that can be read from the buffer. For a PEEK request, the user program must set this field to the same value as first. For an ALTER request involving a message, the user program enters the beginning address of the message.

fnt ord — QAC enters the FNT ordinal of a file when the file is attached from a queue.

Field	Description
ejt/qft ord	The user program enters the QFT ordinal of the file or the EJT ordinal of the job if QAC is to process only the file or job having this ordinal. QAC enters the QFT ordinal of a file when the file is attached during a GET request. For an initial PEEK request, the user program must set this field to zero. QAC enters the QFT or EJT ordinal of the file or job currently being examined if a PEEK request is interrupted because the QAC buffer is full. QAC enters a value of zero when the request is complete.
queue	QAC enters a value indicating the queue in which a file was found for a GET request, or the remaining queues to be examined for a PEEK request. The user program must set this field to zero for an initial PEEK request and must not alter its contents when issuing continuation calls. QAC enters a value of zero upon successful completion of a PEEK request.

Bit(s)	Queue
0	Input queue.
1	Executing job queue.
2	Print queue.
3	Punch queue.
4	Plot queue.
5	Wait queue.
6-7	Reserved for Control Data.
8	Reserved for installations.
9-11	Reserved for Control Data.

limit	The user program enters the limit address (LWA+1) of the buffer to be used by QAC.
dest family	The user program enters the destination family name if QAC is to consider only one family (the program must also set bit 3, addr+7). This represents the family under which you would have to log in to access the file or job at a remote batch terminal. A value of zero selects the default family name. QAC enters the destination family name of the file that is attached during a GET request.
id	The user program enters an ID code if QAC is to consider only one ID code (the user program must also set bit 4, addr+7). QAC enters the ID code of the file attached during a GET request.

Field	Description
origin	The user program enters a value if QAC is to examine files or jobs of only one origin type (the program must also set bit 5, addr+7; if this bit is not set, QAC examines files and/or jobs of all types). On a GET request, the origin type of the attached file is returned in this field.

Origin	Origin type
00	Caller's origin.
01	SYOT (system).
02	BCOT (local batch).
03	RBOT (remote batch).
04	IAOT (interactive).
05-77	Reserved.

dest un	The user program enters the destination user name if QAC is to consider files and/or jobs for only one destination user name (the program must also set bit 3, addr+7). This represents the user name under which you would have to log in to access the file or job at a remote batch terminal. QAC enters the user name of the file that is attached during a GET request.
dest ui	QAC enters the destination user index of the file that is attached during a GET request. This represents the user index under which you would have to log in to access the file or job at a remote batch terminal.
jsn	The user program enters the job sequence name (JSN) if QAC is to consider files and/or jobs for only one JSN (the user program must also set bit 6, addr+7). QAC enters the JSN of the file that is attached during a GET request.
selection flags	The user program sets the appropriate bits in this field to limit QAC's search of jobs or files to specific types. Where noted, these bits are used in conjunction with other fields in the parameter block to specify the type of job or file QAC is to search for.

Bit(s)	Description
0	Specific ordinal (addr+4) and specific queue (addr+10_8).
1	Inhibit duplicate lfn search. If set, QAC will allow a file having the same name as a local file to be attached in a GET request. Otherwise, an attempt to attach such a file causes an error flag in addr+0 to be set and the request is terminated.
2	Include priority=0 in selection. If set, QAC will select files and/or jobs that have a priority of zero, provided they meet other selection criteria.
3	Destination family name (addr+5) or destination user name (addr+6).
4	ID code (addr+5).
5	Origin type (addr+5).
6	JSN (addr+7).
7	Forms code (addr+10_8).
8	Disposition code (addr+10_8).
9	Explicit external characteristics (addr+10_8).
10	Hierarchical external characteristics (addr+10_8).
11	Include ec=0 (any external characteristics) in selection hierarchy (addr+10_8).

Field	Description

Bit(s)	Description
12	Internal characteristics (addr+10_8).
13	Access level (addr+11_8).
14	Destination LID (addr+11_8).
15	List of alternate destination LIDs (addr+11_8).
16	Source LID (addr+11_8).
17	Include LP disposition code in addition to disposition code selected (print files only).
18	Select only recoverable jobs (detached interactive jobs).
19	Enabled destination LID.
20	File size index range specified.
21-32	Reserved for Control Data.
33-35	Reserved for installations.

Bits 3 and 4 may not both be set, and bits 9 and 10 may not both be set. Bit 11 can be set only if bit 9 or bit 10 is set. Bit 17 can be set only if bit 8 is set.

queue sel — The user program sets one or more bits in this field to indicate which queue or queues QAC is to consider.

Bit(s)	Queue
0	Input queue.
1	Executing job queue.
2	Print queue.
3	Punch queue.
4	Plot queue.
5	Wait queue.
6-7	Reserved for Control Data.
8	Reserved for installations.
9-11	Reserved for Control Data.

forms — The user program enters a two-character alphanumeric forms code if QAC is to consider only files with a particular forms code (the program must also set bit 7, addr+7). QAC enters the forms code of the file that is attached during a GET request.

disp — The user program enters a two-character disposition code if QAC is to consider only files with a particular disposition code (the program must also set bit 8, addr+7). QAC enters the disposition code of the file that is attached during a GET request.

disp	Description
IN	Input queue.
LP	Print on any line printer.
LR	Print on 580-12 line printer.
LS	Print on 580-16 line printer.
LT	Print on 580-20 line printer.
NO	Input queue, no output.
PL	Plotter.
PR	Same as LP.
PU	Punch (specific punch type must be entered in the ec field).
TO	Input queue, output to wait queue.
TT	Wait queue.
WT	Same as TT.

Field	Description
ec	The user program enters an external characteristics code if QAC is to consider only files with particular external characteristics (the program must also set bit 9, or 10, addr+7). QAC enters the external characteristics code of the file that is attached during a GET request.

External characteristics of print files:

ec	Description
0	Any external characteristics.
1	Not used.
2	ASCII graphic 48-character set (not supported; provided for NOS/BE compatibility).
3	CDC graphic 48-character set (not supported; provided for NOS/BE compatibility).
4	CDC graphic 63/64-character set.
5	ASCII graphic 63/64-character set.
6	ASCII graphic 95-character set.
7	Reserved.

External characteristics of punch files:

ec	Description
0	System default.
1	System binary.
2	80-column binary.
3	Not used.
4	026 mode.
5	029 mode.
6	ASCII.
7	Reserved.

External characteristics of plot files:

ec	Description
0	Reserved.
1	Transparent 6-bit plotter data.
2	Transparent 8-bit plotter data.
3-6	Not used.
7	Reserved.

If the user program sets bit 9 of addr+7 (explicit external characteristics), QAC searches only for files having the external characteristic selected. If the program sets bit 10 (hierarchical external characteristics), QAC searches for files having compatible external characteristics. The following codes cause QAC to search for files in the order listed.

Field	Description

Hierarchical external characteristics for print files:

ec	Compatible Types
0	ASCII graphic 95-character set, ASCII graphic 63/64-character set, ASCII graphic 48-character set, CDC graphic 63/64-character set, CDC graphic 48-character set.
4	CDC graphic 63/64-character set, CDC graphic 48-character set.
5	ASCII graphic 63/64-character set, ASCII graphic 48-character set.
6	ASCII graphic 95-character set, ASCII graphic 63/64-character set, ASCII graphic 48-character set.

Hierarchical external characteristics for punch files:

ec	Compatible Types
2	80-column binary, system binary.
5	029, 026.

ic — The user program enters an internal characteristics code if QAC is to consider only files with particular internal characteristics (the program must also set bit 12, addr+7). QAC enters the internal characteristics code of the file that is attached during a GET request.

ic	Description
0	6-bit display code.
1	12-bit ASCII.
2	Binary.
3	Reserved.

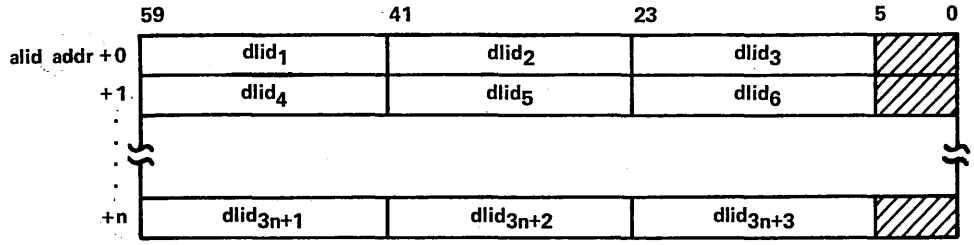
link addr — The user program enters either zero or, in order to process linked QAC requests, the address of the next QAC parameter block. Any nonzero value entered in this field must be a valid address within the user program's field length. Upon completion of the current request, QAC checks this field. If a nonzero value is present, QAC processes the request at address link addr.

slid — The user program enters the source logical identifier (LID) if QAC is to consider only files originating at a particular mainframe. If bit 16 of the selection flags field is set, a value of zero specifies only files without a source LID. QAC enters the source LID of the file that is attached during a GET request.

dlid — The user program enters the destination logical identifier (LID) if QAC is to consider only files with a particular destination LID. If bit 16 of the selection flags field is set, a value of zero specifies only files without a destination LID; a value of *** (474747_8) specifies all files regardless of LID. QAC enters the destination LID of the file that is attached during a GET request.

al lb — The user program enters the access level lower bound if QAC is to consider only files with an access level equal to or greater than a particular value (the program must also set bit 13, addr+7).

Field	Description
al ub	The user program enters the access level upper bound if QAC is to consider only files with an access level equal to or less than a particular value (the program must also set bit 13, addr+7).
alid addr	The user program enters the address of a list of alternate destination LIDs if QAC is to consider only files from that list of destination LIDs. The list of alternate LIDs has the following format:



The list is terminated by the first 18-bit zero field starting at bit 59, 41, or 23.

li	The user program enters the file size index lower bound if QAC is to consider only files with a file size index within a specified range (the program must also set bit 20, addr+7). If li is specified, ui must also be specified, and li must be .LE. ui.
ui	The user program enters the file size index upper bound if QAC is to consider only files with a file size index within a specified range (the program must also set bit 20, addr+7). If ui is specified, li must also be specified, and ui must be .LE. li.

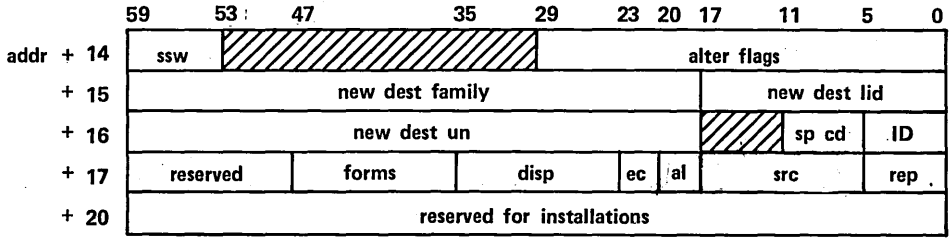
QAC REQUESTS

The following pages describe the three types of QAC requests: ALTER, GET, and PEEK. Each type of request uses a different format for the remainder of the parameter block following addr+13_8.

ALTER REQUEST

The ALTER request changes characteristics of queued files or executing jobs. The alter flags in addr+14_8 determine which characteristics the ALTER request is to change. QAC cannot alter the calling job.

The format of the remainder of the parameter block for an ALTER request is as follows:



Field	Description
ssw	The user program enters bit values to which the sense switch settings of jobs are to be altered. Bits 54 through 59 correspond to sense switches 1 through 6 respectively. If bit 10 of addr+14_8 (turn sense switches on) is set, the sense switches whose corresponding bits in this field are set to one are changed to one for the selected jobs. If bit 11 of addr+14_8 (turn sense switches off) is set, the sense switches whose corresponding bits in this field are set to zero are changed to zero for the selected jobs. Refer to the ONSW and OFFSW macro descriptions in section 6 for descriptions of the individual sense switches.
alter flags	The user sets the appropriate bits in this field to specify the information that QAC is to alter.

Bit(s)	Description
0	Change remote family name, user name.
1	Change local batch identifier.
2	Reserved for Control Data.
3	Alter forms code.
4	Alter spacing code.
5	Alter disposition code.
6	Alter external characteristics.
7	Alter repeat count.
8	Alter access level.
9	Abort job or evict file.
10	Turn sense switches on.
11	Turn sense switches off.
12	Clear pause bit.
13	Send comment to dayfile.
14	Send comment from operator.
15	Indicate message available for aborted job.
16	Set pause bit.
17	Alter destination logical identifier.
18	Kill job.
19	Alter service class.
20-26	Reserved for Control Data.
27-29	Reserved for installations.

Field	Description
	Bit 15 can be set only if bit 9 is set. Only one of bits 13, 14, and 15 can be set. If bit 13, 14, or 15 is set, the program must enter a message in the QAC buffer defined by the fields FIRST, IN, OUT, and LIMIT.
new dest family	The user program enters the new destination family name for the files and/or jobs. The program must also set bit 0, addr+14_8. QAC changes the destination family and routes the files/jobs to remote batch. The specified family must be known to the system but need not be active. If this field is selected and has a value of zero, QAC uses the family of the requesting job's control point.
new dest lid	The user program enters the new destination logical identifier (LID) for selected input and/or output files. The program must also set bit 16, addr+14_8.
new dest un	The user program enters the new destination user name for the files and/or jobs. The program must also set bit 0, addr+14_8. QAC changes the destination user name and routes the files/jobs to remote batch. If this field is selected and has a value of zero, QAC uses the user number associated with the job in which the user program is executing.
sp cd	The user program enters the new spacing code for the files. This field affects only print files. The program must also set bit 4, addr+14_8. Values of 0 through 77_8 are accepted; if the value entered is not defined for the installation, the released default value (0) is used. For more information on spacing codes, refer to the NOS 2 Analysis Handbook.
id	The user program enters the identifier for the local batch queue to which the files and/or jobs are to be routed. The program must also set bit 1, addr+14_8.
forms	The user program enters the new forms code for the files. This field affects only output files. The program must also set bit 3, addr+14_8. This field may contain any two alphanumeric characters or zero.
disp	The user program enters the new disposition code for the files. This field affects only output files. The program must also set bit 5, addr+14_8. Valid disposition codes are listed in the description of the disp field of addr+10_8.
ec	The user program enters the new external characteristics code for the files. This field affects only output files. The user program must also set bit 6, addr+14_8. Valid external characteristics codes are listed in the description of the ec field of addr+10_8. A value of zero for a punch file selects the installation-defined default punch mode.
al	The user program enters the new access level for the files. The program must also set bit 8, addr+14_8. This field may contain any value from 0 through 7. (You can convert a symbolic access level name to a numerical value, or vice versa, using system OPL common deck COMCVLC.)

Field	Description
src	The user program enters the new service class for files in the input queue and for executing jobs. The program must also set bit 19, addr+14_8. This field does not affect output files. The service class is specified as the first two characters in the display code of one of the service class mnemonics listed in appendix E.
rep	The user program enters the new repeat count for output files. The program must also set bit 7, addr+14_8. The maximum value allowed is 37_8.

Example:

The following program clears the pause bit in all executing jobs submitted under the same user name as the program (except the job in which the program is executing). The program also enters the message PROGRAM RESTART HAS CLEARED PAUSE BIT in the user dayfile of each job. This program uses the SYSTEM macro (refer to section 11) to issue the QAC request.

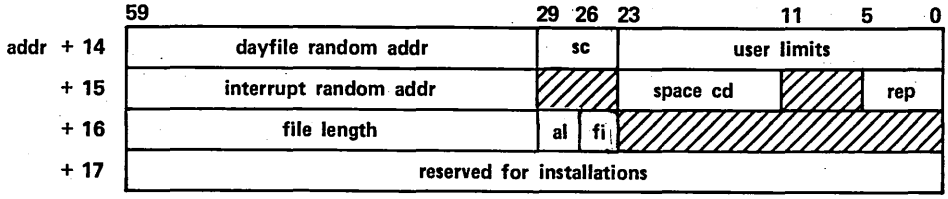
```
          IDENT  RESTART
          ENTRY  RESTART

**        MESSAGE BUFFER

MSG       DATA   C*PROGRAM RESTART HAS CLEARED PAUSE BIT*
TAG1      EQU    *-MSG

**        QAC PARAMETER BLOCK

BLOCK     BSS    0
          VFD    59/1,1/0              ALTER FUNCTION CODE
          VFD    36/0,6/12,18/MSG      MESSAGE BUFFER, FIRST
          VFD    42/0,18/MSG+TAG1            "         , IN
          VFD    42/0,18/MSG                 "         , OUT
          VFD    42/0,18/MSG+TAG1+1          "         , LIMIT
          VFD    60/0
          VFD    60/0
          VFD    60/0
          VFD    12/2,48/0             SELECT EXECUTING JOB QUEUE
          VFD    60/0
          VFD    60/0
          VFD    60/0
          VFD    60/220000B            CLEAR PAUSE BIT AND SEND
          VFD    60/0                    COMMENT TO DAYFILE
          VFD    60/0
          VFD    60/0
          VFD    60/0

**        MAIN PROGRAM

RESTART   SYSTEM QAC,R,BLOCK            CALL QAC
          ENDRUN


          END    RESTART
```

GET REQUEST

The GET request attaches one of your queued files to your job in read mode.

The attached file can be returned to the queue with a DSP call or a ROUTE macro. The user program can drop the attached file by issuing a RETURN macro. If the file remains attached when the job terminates, the file is returned to the queue from which it was attached.

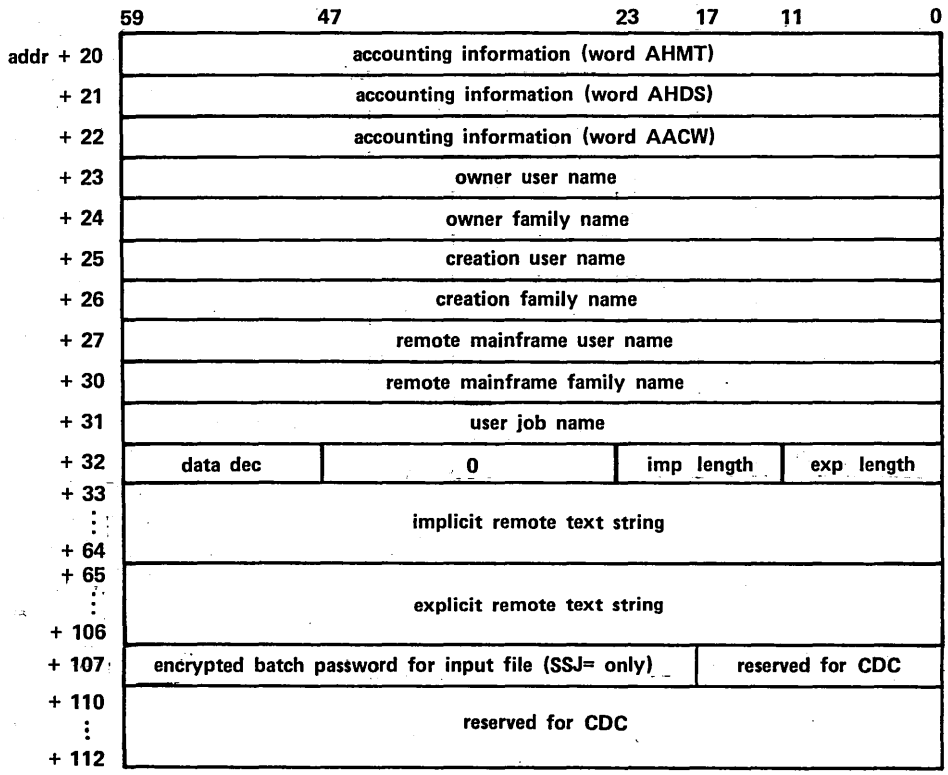
The format of the remainder of the parameter block for a GET request is as follows:



Field	Description
dayfile random addr	QAC enters the random address of the first PRU of the queued file's dayfile. If the file does not contain a dayfile, QAC enters a value of zero. This field applies only to print files.
sc	QAC enters the service class of the file. Refer to appendix E for a list of valid service classes and their corresponding 6-bit octal values.
user limits	QAC enters your authorized resource limit that applies to the type of device to which the file is routed. A value of 77777777_8 is defined as unlimited. This field applies only to output files.
interrupt random addr	If a GET request was previously performed on a file and the file was later returned to the queue, QAC enters file position from the time the file was returned to the queue. This value can be used to reposition the file.
space cd	QAC enters the spacing code of the queued file. This field applies only to print files.
rep	QAC returns the repeat count specified for the queued file. This field applies only to output files.
file length	QAC enters the length of the file in PRUs.
al	QAC enters the security access level of the file.
fi	QAC enters the file size index of the file.

An extended version of the GET function can be selected by entering a length of 112_8 in the len field in word addr+1 of the QAC parameter block. The extended GET function returns additional accounting information and control information for remote host files.

QAC returns the following information to addr+20_8 through addr+106_8 of the QAC parameter block for an extended GET request. Extended GET request parameter block (len field of addr+1 of QAC parameter block = 112_8):



Field	Description
accounting information	Accounting words AHMT, AHDS, and AACW for the file. Refer to the NOS 2 Administration Handbook for descriptions of these words.
owner user name	The user name of the owner of the file.
owner family name	The family name of the owner of the file.
creation user name	The user name of the creator of the file.
creation family name	The family name of the creator of the file.
remote mainframe user name	For remote host files, the user name under which the file is accessed on a remote mainframe. For other files, this field has a value of zero.

Field	Description
remote mainframe family name	For remote host files, the family name under which the file is accessed on a remote mainframe. For other files, this field has a value of zero.
user job name	The user job name for the file.
data dec	Data declaration field from DSP parameter block (files routed to remote mainframe using extended DSP request only).
imp length	Character length of implicit remote text string (remote host files only).
exp length	Character length of explicit remote text string (remote host files only).
implicit remote text string	For remote host files, a string of up to 400_8 characters that contains parameters for the operating system under which the file was created. For other files, this field has a value of zero.
explicit remote text string	For remote host files, an optional string of up to 400_8 characters that form a job command in the job control language of the operating system of the mainframe that is to process the file. For other files, this field has a value of zero.

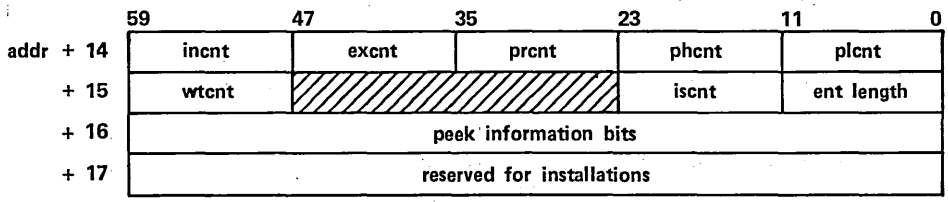
PEEK REQUEST

The PEEK request examines selected queued files or executing jobs, and returns information about those files and/or jobs to the QAC buffer. The user program specifies which files/jobs QAC is to consider by setting the appropriate selection flags and their corresponding fields in words addr+0 through addr+13_8 of the parameter block and selects the information about those files/jobs that QAC is to return.

You can use the PEEK request to obtain a count of queued files and executing jobs that belong to you. To do this, include a PEEK request in your program without setting any peek information bits; QAC will enter the number of queued files and executing jobs for your user name in words addr+14_8 and addr+15_8.

When the QAC buffer is filled with PEEK reply information during a PEEK request, QAC enters the QFT or EJT ordinal of the file or job last examined in the ejt/qft ord field of addr+4 and enters the queue last examined in the queue field of addr+4. To issue a continuation call, the user program clears the completion bit (addr+0) and empties the QAC buffer by setting out (addr+3) to the value of in (addr+2). The program then reissues the RA+1 call. QAC continues its search from the file or job at which it was interrupted.

The format of the remainder of the parameter block for a PEEK request is as follows:

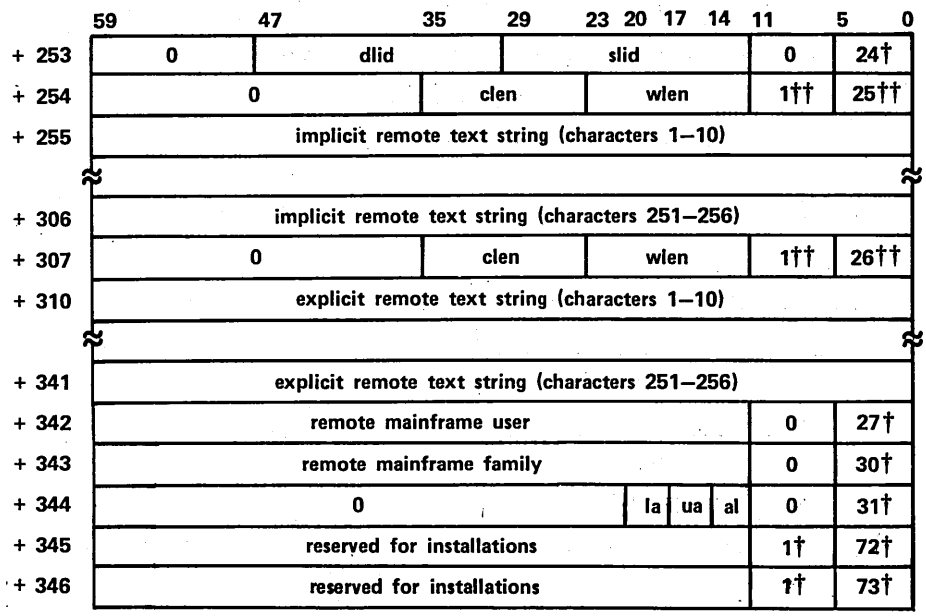


Field	Description
incnt	If no peek information bits are set (addr+16$_8$), QAC enters the number of files in the input queue that are eligible for selection. In order for any input queue files to be eligible for selection, the user program must set bit 0 of the queue sel field, addr+10$_8$. If any peek information bits are set, the user program must enter the maximum number of files to be selected from the input queue. When the PEEK request is completed, QAC enters the actual number of input queue files selected.
excnt	This field is used in the same manner as the incnt field, except that it applies to jobs in the executing job queue. In order for any jobs in the executing job queue to be eligible for selection, the user program must set bit 1 of the queue sel field, addr+10$_8$.
prcnt	This field is used in the same manner as the incnt field, except that it applies to files in the print queue. In order for any print queue files to be eligible for selection, the user program must set bit 2 of the queue sel field, addr+10$_8$.
phcnt	This field is used in the same manner as the incnt field, except that it applies to files in the punch queue. In order for any punch queue files to be eligible for selection, the user program must set bit 3 of the queue sel field, addr+10$_8$.
plcnt	This field is used in the same manner as the incnt field, except that it applies to files in the plot queue. In order for any plot queue files to be eligible for selection, the user program must set bit 4 of the queue sel field, addr+10$_8$.
wtcnt	This field is used in the same manner as the incnt field, except that it applies to files in the wait queue. In order for any wait queue files to be eligible for selection, the user program must set bit 5 of the queue sel field, addr+10$_8$.
iscnt	This field is reserved for installations.
ent length	QAC enters the length, in words, of each reply in the peek reply buffer. If no peek information bits are set, QAC enters a value of zero. QAC creates a block of replies that all have the same length; any words in a particular reply that are not defined for the queue type of the reply contain zeros.
peek information bits	The bits in this word correspond to the octal numbers in the lower 6 bits of words in the peek reply block. Each bit that is set in this word causes QAC to return the information shown in the corresponding word of the peek reply block. Certain bits select information contained in more than one word in the reply block: bit 14 selects message buffer one for the control point area (MS1W), bit 15 selects message buffer two for the control point area (MS2W), bit 18 selects dayfile messages, bit 19 selects the command buffer, and bit 24 selects the remote text string. Symbols for these peek information bits are defined in the system OPL common deck COMSQAC.

For each file or job that satisfies the selection criteria specified in the QAC parameter block, QAC returns a PEEK reply block to the QAC buffer. Each PEEK reply block has the following format.

	59–53	47–44	41–35	29–23	17–11	5–0
out + 0	jsn		ordinal	queue	0	01†
+ 1	sc	ot forms	disp	ec	ic 0	02†
+ 2	priority	fl-em	fl-cm	0	0	03†
+ 3	destination family				0 0	04†
+ 4	destination un				id 0	05†
+ 5	creation family				0 0	06†
+ 6	creation un				cid 0	07†
+ 7	dfo	destination ui	cfo	creation ui	0	10†
+ 10	user job name				0 0	11†
+ 11	space cd	I 0 rep	account limit		0	12†
+ 12	esto	creation date + time			0	13†
+ 13	file length		creation jsn		0	14†
+ 14	ejt stat	0 c	scheduling field		0	15†
+ 15	0			len1	1††	16††
+ 16	MS1W characters 1–10					
+ 17	MS1W characters 11–20					
+ 20	MS1W characters 21–30					
+ 21	MS1W characters 31–40					
+ 22	MS1W characters 41–50					
+ 23	0			len2	1††	17††
+ 24	MS2W characters 1–10					
+ 25	MS2W characters 11–20					
+ 26	MS2W characters 21–30					
+ 27	SRU limit				0	20†
+ 30	SRUs accumulated				0	21†
+ 31	0		alen3	len3	1	22††
+ 32	dayfile messages (first word)					
≈	≈					≈
+ 141	dayfile messages (word len3)					
+ 142	0		alen4	len4	1	23††
+ 143	commands (first word)					
≈	≈					≈
+ 252	commands (word len4)					

†This field has a value of zero if the rest of the word contains all zeros.
††This field has a value of zero if the following word contains all zeros.



Field	Description
jsn	The job sequence name of the selected job or file.
ordinal	The EJT or QFT ordinal of the selected job or file.
queue	The queue in which the job or file is located.

Bit(s)	Queue
0	Input queue.
1	Executing job queue.
2	Print queue.
3	Punch queue.
4	Plot queue.
5	Wait queue.
6-7	Reserved for Control Data.
8	Reserved for installations.
9-11	Reserved for Control Data.

sc	The service class of the job or file.
ot	The origin type of the job or file.
forms	The forms code for the file. Does not apply to jobs in the executing queue or files in the input queue.

†This field has a value of zero if the rest of the word contains all zeros.
††This field has a value of zero if the following word contains all zeros.

Field	Description
disp	The disposition code assigned to the file. Does not apply to jobs.
ec	The external characteristics of the file. Does not apply to entries in the executing queue or the input queue.
ic	The internal characteristics of the file. Does not apply to entries in the executing queue or the input queue.
priority	The priority of the file at the time the PEEK request took place. Does not apply to executing jobs.
fl-em	The current extended memory field length for the job. Applies only to jobs in the executing queue.
fl-cm	The current central memory field length for the job. Applies only to jobs in the executing queue.
destination family	The destination family for the file, if the file is not routed to a local batch device. This represents the family under which you would have to log on to a remote batch terminal in order to access the file. A value of zero is entered if the file is routed to a local batch device.
destination un	The destination user name for the file, if the file is not routed to a local batch device. This represents the user name under which you would have to log on to a remote batch terminal in order to access the file. A value of zero is entered if the file is routed to a local batch device.
id	The destination batch device ID, if the file is routed to a local batch device; otherwise, zero.
creation family	The family on which the file or job was created.
creation un	The user name under which the file or job was created.
cid	The device ID of the device at which the file or job was created, if it was created at a local batch device; otherwise, zero.
dfo	The family ordinal corresponding to the destination family in word out+3. A value of zero indicates no family ordinal.
destination ui	The user index corresponding to the destination user name in word out+4. A value of zero indicates no user index.
cfo	The family ordinal corresponding to the creation family in word out+5. A value of zero indicates no family ordinal.
creation ui	The user index corresponding to the creation user name in word out+6. A value of zero indicates no user index.
user job name	The user job name associated with the file or job.

Field	Description
space cd	The spacing code for the entry. Does not apply to jobs in the executing queue or files in the input queue.
I	This bit is set if the file is interrupted. Applies only to output files.
rep	The repeat count for the entry. Does not apply to jobs in the executing queue or files in the input queue.
account limit	The user's validated resource limit that applies to the type of device to which the file or job is routed. A value of 77777777_8 is defined as unlimited. Does not apply to jobs in the executing queue or files in the input queue.
esto	The EST ordinal of the device on which the file resides. Does not apply to jobs in the executing queue.
creation date+time	The packed date and time when the file was first queued. The six 6-bit octal fields contain the following: year minus 1970, month, day, hour, minute, and second.
file length	The length of the file in PRUs. Does not apply to jobs in the executing queue.
creation jsn	The job sequence name of the job that created the file or job.
ejt stat	The status field of the EJT entry. Applies only to executing jobs. Bits 1 through 5 contain the EJT job status of the job; bits 7 through 10 contain the EJT connection status of the job. Refer to appendix E for a listing of valid EJT status codes and their meanings.
c	CPU status. Applies only to executing jobs.

Value	Description
0	Null status.
1	A status (job active in CPU 0).
2	B status (job active in CPU 1).
3	Not used.
4	W status (job waiting for CPU).
5	X status (job waiting for recall).
6	I status (job waiting for PP drop).
7	Not used.

scheduling field	The contents of this field depend on the EJT status of the job. Applies only to jobs in the execution queue. Refer to appendix E for a description of the scheduling field associated with each EJT status code.
lenl	The length of MS1W in words. This field always has a value of five.
MS1W	The message in message buffer 1 of the job's control point area. The message can contain up to 50 characters; if the message is less than 50 characters long, it is terminated by the standard display coded line terminator sequence.

Field	Description
len2	The length of MS2W in words. This field always has a value of three.
MS2W	The message in message buffer 2 of the job's control point area. The message can contain up to 30 characters; if the message is less than 30 characters long, it is terminated by the standard display coded line terminator sequence.
SRU limit	The job's SRU limit (from the job control point area).
SRUs accumulated	The current number of SRUs accumulated by the job (from the job's control point area).
alen3	The number of nonzero words in the dayfile message entry.
len3	The length of the dayfile message entry in words. This field always has a value of 114_8.
dayfile messages	QAC enters a copy of the job's dayfile buffer. Dayfile messages have the standard display coded line format.
alen4	The number of nonzero words in the commands entry.
len4	The length of the commands in words. This field always has a value of 110_8.
commands	QAC enters a copy of the job's command buffer. Commands already executed and commands not yet read into the buffer do not appear. Commands have the standard display coded line format.
dlid	Destination logical identifier for the file.
slid	Source logical identifier for the file.
clen	The length of the implicit or explicit remote text string in characters.
wlen	The length of the implicit or explicit remote text string in words.
implicit remote text string	A string of up to 400_8 characters that contains parameters for the operating system under which the file was created. This string is present only for remote host files.
explicit remote text string	A string of up to 400_8 characters that form a job control command in the job control language of the operating system of the mainframe that is to process the file. This string is present only for remote host files.
remote mainframe user	The user name under which you would have to log on to a remote batch terminal in order to access the file at the destination mainframe. This word is present only for remote host files.
remote mainframe family	The family name under which you would have to log on to a remote batch terminal in order to access the file at the destination mainframe. This word is present only for remote host files.

Field	Description
la	The lower access level limit of the input file or job.
ua	The upper access level limit of the input file or job.
al	The current access level of the file or job.

Example:

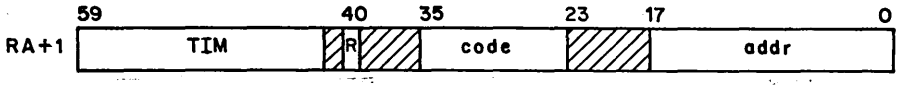
The following program locates up to 10 executing jobs that were submitted under the same user name as the program. All information about those jobs is stored in the PEEK reply buffer and is dumped. This program uses the SYSTEM macro (refer to section 11) to issue the QAC request.

```
          IDENT  PEEKJOB
          ENTRY  MAIN

**        QAC PARAMETER BLOCK

BLOCK     BSS    0
          VFD    59/3,1/0                  PEEK FUNCTION CODE
          VFD    36/0,6/11,18/REPLY        PEEK REPLY BUFFER, FIRST
          VFD    42/0,18/REPLY                    "       , IN
          VFD    42/0,18/REPLY                    "       , OUT
          VFD    42/0,18/REPLY+1000               "       , LIMIT
          VFD    60/0
          VFD    60/0
          VFD    60/0
          VFD    12/2,48/0                 SELECT EXECUTING JOB QUEUE
          VFD    60/0
          VFD    60/0
          VFD    60/0
          VFD    12/0,12/10,36/0           SELECT UP TO 10 ENTRIES
          VFD    60/0
          VFD    37/0,23/-0                SELECT ALL INFORMATION
          VFD    60/0

**        MAIN PROGRAM

MAIN      SYSTEM QAC,R,BLOCK               CALL QAC
          SYSTEM DMD,R,BLOCK+15,BLOCK      DUMP UPDATED PARAMETER BLOCK
          SYSTEM DMD,R,REPLY+1000,REPLY    DUMP PEEK REPLY BUFFER
          ENDRUN

**        PEEK REPLY BUFFER

REPLY     BSSZ   1001                      REPLY BUFFER

          END    MAIN
```

SYSTEM TIME/DATE REQUESTS 9

NOS provides several system requests associated with the time and date. These requests are processed directly by CPUMTR. Unless otherwise noted, the only common deck required for absolute assemblies is COMCSYS.

The format of the RA+1 call to the time function processor is:



R Auto recall bit.

code Function code as follows:

Code	Macro	Description
0000	TIME	Accumulated CP time used by the job.
0001	DATE	Current date in display code.
0002	CLOCK	Current time of day in display code.
0003	JDATE	Current Julian date.
0005	RTIME	Elapsed real time since last deadstart.
0006	PDATE	Current date and time in packed binary form.
0007	STIME	Accumulated system resource units.
0011	HTIME	Accumulated clock cycle count for control point.
0012	HTIME	Accumulated clock cycle count since deadstart.

addr Address of a reserved one-word block that will receive the reply.

The EDATE and ETIME macros described in this section do not have an RA+1 call. They are processed by common deck COMCEDT and are used to convert the format of a date and time returned by a previous macro call.

CLOCK

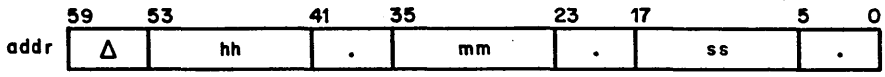
In response to the CLOCK function, the system returns the current time of day in display code to location addr.

Macro format:

Location	Operation	Variable
	CLOCK	addr

addr Address of reply word.

Reply word format (display code):



Δ Space (display code 55_8).

hh Hours.

mm Minutes.

ss Seconds.

DATE

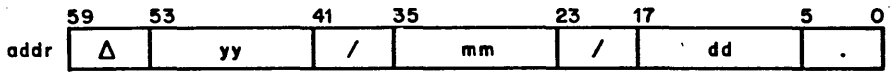
In response to the DATE function, the system returns the current date in display code format to location addr.

Macro format:

Location	Operation	Variable
	DATE	addr

addr Address of reply word.

Reply word format (display code):



Δ Space (display code 55_8).

yy Year (last two digits).

mm Month.

dd Day.

EDATE

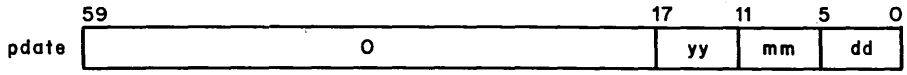
EDATE converts the packed date in address pdate to display code and enters the converted date in register X6.

Macro format:†

Location	Operation	Variable
	EDATE	pdate

pdate Register or address containing packed date to be converted.

Format of address or register containing packed date:

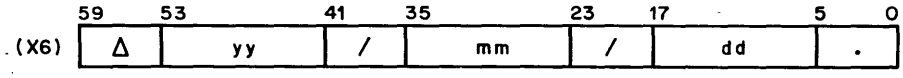


yy Year minus 1970.

mm Month.

dd Day.

Upon completion, register X6 contains the following (display code):



Δ Space (display code 55_8).

yy Year (last two digits).

mm Month.

dd Day.

This macro requires the common decks COMCEDT and COMCCDD.

†This macro is not available in SYSTEXT. The user program must specify systems text PSSTEXT or call common deck COMCMAC (refer to appendix F).

ETIME

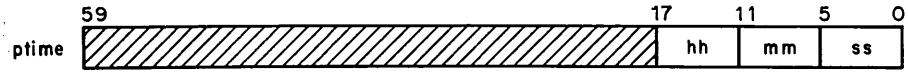
ETIME converts the packed time in address ptime to display code and enters the converted time in register X6.

Macro format:†

Location	Operation	Variable
	ETIME	ptime

ptime Register or address containing packed time to be converted.

Format of address or register containing packed time:

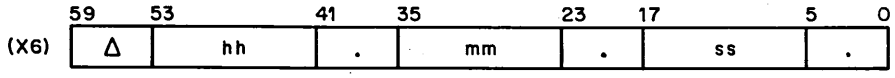


hh Hours.

mm Minutes.

ss Seconds.

Upon completion, register X6 contains the following (display code):



Δ Space (display code 55_8).

hh Hours.

mm Minutes.

ss Seconds.

This macro requires the common decks COMCEDT and COMCCDD.

† This macro is not available in SYSTEXT. The user program must specify systems text PSSTEXT or call common deck COMCMAC (refer to appendix F).

HTIME

HTIME returns the model 176 accumulated CPU clock cycle count (if available) either for the current job or since the last system deadstart. A clock cycle on the model 176 is 27.5 nanoseconds.

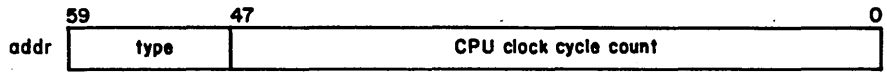
Macro format:

Location	Operation	Variable
	HTIME	addr,op

addr — Address of reply word.

op — Option desired; if 0, accumulated clock cycle count for the current job is returned (function code 11); if 1, accumulated clock cycle count since the last system deadstart is returned (function code 12).

Reply word format:



type — If the model 176 CPU clock cycle count is available, type is 2000_8.
If the model 176 CPU clock cycle count is not available, type is 6000_8.

JDATE

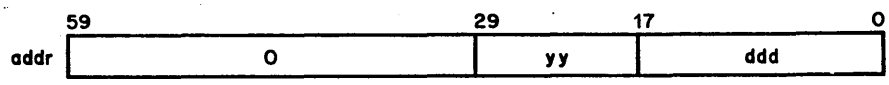
JDATE returns the current Julian date, in display code, to location addr.

Macro format:

Location	Operation	Variable
	JDATE	addr

addr — Address of reply word.

Reply word format (display code):



yy — Julian year.

ddd — Julian day.

PDATE

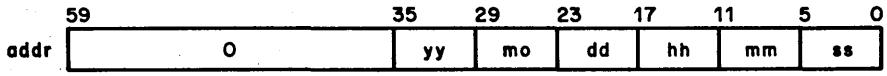
PDATE returns the current date and time in binary packed format. The user program can unpack the parameters or use the EDATE and ETIME macros to do the unpacking.

Macro format:

Location	Operation	Variable
	PDATE	addr

addr Address of reply word.

Reply word format:



yy Year minus 1970.

mo Month.

dd Day.

hh Hours.

mm Minutes.

ss Seconds.

RTIME

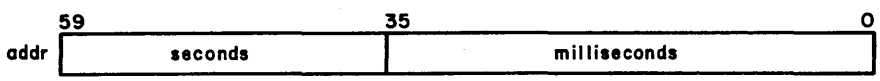
RTIME returns the real-time clock reading to location addr. This is the elapsed time since deadstart.

Macro format:

Location	Operation	Variable
	RTIME	addr

addr Address of reply word.

Reply word format:



seconds Time since deadstart in seconds.

milliseconds Time since deadstart in milliseconds.

STIME

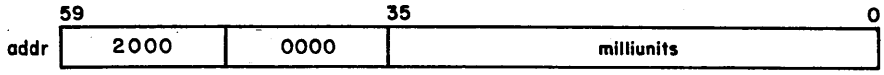
STIME returns the job's accumulated system resource units (SRUs). Refer to the NOS 2 Analysis Handbook for a description of SRUs.

Macro format:

Location	Operation	Variable
	STIME	addr

addr Address of reply word.

Reply word format:



milliunits Accumulated system resource units/1000.

TIME

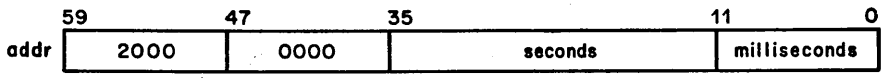
TIME returns the accumulated central processor time used by the job to location addr.

Macro format:

Location	Operation	Variable
	TIME	addr

addr Address of reply word.

Reply word format:



seconds Accumulated central processor time in seconds.

milliseconds Accumulated central processor time in milliseconds (modulo 1000).

NOTE

In order to use the values returned by the TIME macro to determine total accumulated central processor time, the user program must perform the following conversion:

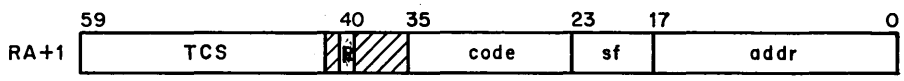
Total time (milliseconds) = (seconds x 1000) + milliseconds.

Attempting to use bits 35 through 0 as a single quantity will produce meaningless results.

TRANSLATE CONTROL STATEMENT

Translate control statement (TCS) processes user requests to read a command from or to place a command into the stream. The only common deck required for absolute assembly is COMCSYS.

The format of the RA+1 call is:



R Auto recall bit.

code Function code:

Code	Macro
004	CONTROL
005	EXCST

sf Subfunction code for the CONTROL macro. This field is not used for the EXCST macro.

sf	Action
00	Read the next command and advance the command pointer.
01	Read the next command only if it does not call for a load from a local file or from a global library set. Do not advance the command pointer.
02	Read the next command even if it calls for a load from a local file or from a global library set. Do not advance the command pointer.
4x	If the next command being read should be processed in product set format, sf should be set to 4x, where x can be 0, 1, or 2, corresponding to one of the above options. If sf is not set to 4x, command parameters are to be processed in NOS format.

addr First word address of the buffer in which the command is to be stored or from which the command is to be read.

CONTROL (004)

The CONTROL macro allows the user program to read the next command in the command stream and transfer it to a specific address. The command is checked for syntax errors, and all parameters are stored as if a program load had actually taken place.

If no command exists, binary zero words are returned to the buffer. If the next command is of the form:

```
*comments
```

the command is returned to the buffer if the comments string ends with a period or right parenthesis. Otherwise, the binary zero words are returned to the buffer.

Macro format:

Location	Operation	Variable
	CONTROL	addr,rss,lf,psf

addr — First word address of the buffer in which the next command is to be stored. The user program should allow room for eight words (80 characters).

rss — If rss is specified (any value may be used), the command pointer is not advanced. This allows the user program to determine what the next command is and still allow it to be processed. If rss is not specified, the command pointer is advanced as if the command had been processed.

lf — If rss is not specified, lf is ignored. If the following are true:

- rss is specified,
- lf is not specified,
- the command to be read calls for a load from a local file or from a global library set,

the command will not be read. If rss and lf are both specified (any value may be used for lf), the command will always be read. If the command calls for a load from a local file or from a global library set, bit 17 or RA+64_8 will be set.

psf — If psf is specified (any value may be used), parameters are unpacked in product set format.

The command parameters are stored in the job communication area (refer to figure E-1) beginning at ARGR (RA+2) through RA+n (n cannot exceed 63_8). These parameters are terminated by a zero word. The number of parameters stored in the job communication area is placed in the lower 18 bits of ACTR ($RA+64_8$). The name of the command is placed in bits 59 through 18 of PGNR ($RA+64_8$). The parameters are stored in either operating system or product set format, depending on psf. The command image, less any label or prefix field, is stored beginning at addr.

If the parameters are stored in operating system format, the comma, period, and right parenthesis are stored as zero. For all parameters and valid separators except the comma, their display code equivalent is stored.

If the parameters are stored in product set format, separators and terminators are stored as follows:

Character	Code (Octal)
,	1
=	2
/	3
(	4
+	5
-	6
;	10
) or .	17
Others	16

The manner in which CONTROL stores the command is identical to the way the system normally processes commands, except that the command image is stored at the specified address. When the system processes a command, the command image is stored beginning at CCDR ($RA+70_8$). Calling the CONTROL macro does not affect the contents of CCDR.

Example:

The following program reads the next command in the command stream (advancing the command pointer), stores the parameters in operating system format, and transfers the command image to the specified buffer.

```
            IDENT   CMND
            ENTRY   CMND

CMBUF       BSS     8

CMND        CONTROL CMBUF
            SYSTEM  DMD,R,63B,2
            SYSTEM  DMD,R,CMBUF+7,CMBUF
            ENDRUN

            END     CMND
```

When the program is executed from a batch job or in a procedure file and is followed in the command stream by:

```
TEST,A=B,C=D,E,F.
```

the contents of central memory are displayed as follows:

```
ARGR    RA+2    0100    0000    0000    0000    0054    A       =
          +3    0200    0000    0000    0000    0000    B
          +4    0300    0000    0000    0000    0054    C       =
          +5    0400    0000    0000    0000    0000    D
          +6    0500    0000    0000    0000    0000    E
          +7    0600    0000    0000    0000    0000    F
          +10   0000    0000    0000    0000    0000
                .
                .
                .
CMBUF     +0    2405    2324    5601    5402    5603    TEST,A=B,C
          +1    5404    5605    5606    5700    0000    =D,E,F.
                .
                .
                .
```

If the macro call is changed to

```
CONTROL CMBUF,,,PSF
```

the command parameters are stored in product set format and the contents of locations RA+2 through $RA+10_8$ appear as follows:

```
ARGR    RA+2    0100    0000    0000    0000    0002    A       B
          +3    0200    0000    0000    0000    0001    B       A
          +4    0300    0000    0000    0000    0002    C       B
          +5    0400    0000    0000    0000    0001    D       A
          +6    0500    0000    0000    0000    0001    E       A
          +7    0600    0000    0000    0000    0017    F       0
          +10   0000    0000    0000    0000    0000
```

EXCST (005)

With the EXCST macro the user program specifies a buffer containing a command. Control is transferred from the calling program to the system, which reads the command, places it in the command stream, and processes it. Control is not returned to the calling program. The command must conform to NOS command format conventions described in Volume 3, System Commands.

Macro format:†

Location	Operation	Variable
	EXCST	addr

addr — First word address of the buffer containing the command to be processed. The system begins reading at addr and continues until the end of statement (zero byte) or end of buffer (80 characters) is reached. The command must be left-justified with zero fill.

CHECKPOINT/RESTART

A job may be terminated at any time as the result of system, operator, or program error. For some jobs it becomes more advantageous to accept the overhead of checkpoint procedures than to run the risk of losing the entire job output. The checkpoint/restart feature is implemented through the CKP command or CHECKPT macro and the RESTART command. Refer to Volume 3, System Commands, for discussions of the CKP and RESTART commands.

CHECKPT

The CHECKPT macro is used for taking checkpoint dumps. The dump is written on the tape or mass storage checkpoint file specified on a REQUEST, ASSIGN, or LABEL command or REQUEST or LABEL macro. For a general description of checkpoint dumps, refer to Volume 3, System Commands. The CHECKPT macro provides you with greater control than the CKP command in specifying the type of copy to be performed.

NOTE

A checkpoint is not allowed in a program that has issued a PROTECT macro selecting user file privacy, in a program that has issued a SETSSM macro, or in a program that has been loaded from an execute-only file.

† This macro is not available in SYSTEXT. The user program must specify systems text PSSTEXT or call common deck COMCMAC (refer to appendix F).

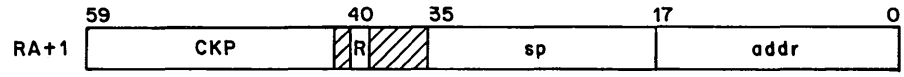
Macro format:†

Location	Operation	Variable
	CHECKPT	addr,sp

addr — Address of the parameter list identifying the files to be checkpointed.

sp — Flag indicating whether or not all files assigned to the job are to be checkpointed. If sp is specified (any value may be used), only those files specified in the parameter block at address addr will be checkpointed. If the sp field is not specified, all local files will be checkpointed.

The format of the RA+1 call is:

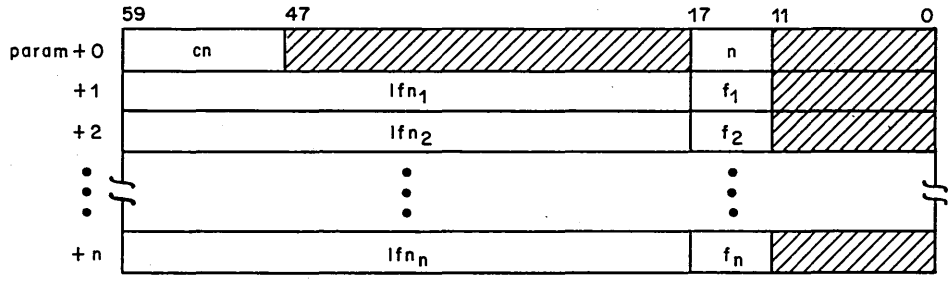


R — Auto recall bit.

sp — Flag indicating whether or not all files assigned to the job are to be checkpointed. If sp is specified (any value may be used), only those files specified in the parameter block at address addr will be checkpointed. If the sp field is not specified, all local files will be checkpointed.

addr — Address of the parameter block identifying the files to be checkpointed.

Parameter list format:



cn — Latest checkpoint number.

n — Octal number of entries in the parameter list ($0 \leq n \leq 77_8$).

† This macro is not available in SYSTEXT. The user program must specify systems text PSSTEXT or call common deck COMCMAC (refer to appendix F).

lfn_i Identifies file to be checkpointed; lfn_i is left-justified.

f_i Specifies the manner in which lfn_i is to be copied:

f_i	Description
0	The mass storage file is copied from the BOI to its position at checkpoint; only that portion is available for restart. RESTART positions the file at the latter point.
1	The mass storage file is copied from its position at checkpoint to the EOI; only that portion is available for restart. RESTART positions the file at the former point.
2	The entire mass storage file is copied. RESTART sets the file to its position at checkpoint time.
3	The last operation on the file determines how the mass storage file is copied.

Last Operation	f Selected
Write	f_i=0
Read (EOI detected)	No copy
Read (EOI not detected)	f_i=2

4	The information table associated with the file is copied but the file itself is not copied. The information table contains FNT/FST information and the random address of the mass storage file or the block count of the magnetic tape file. RESTART retrieves the file in some cases (refer to RESTART) and sets it to its position at checkpoint time. If f_i=4 and the file is a mass storage file, RESTART assumes it is an indirect access file and issues a GET macro to obtain a working copy.

If a value other than 4 is specified for a magnetic tape file, a warning message is issued and the file is processed as if f_i =4 had been specified. f_i =4 is also forced for execute-only files.

The following list shows the type of operation CHECKPT performs for the different values of sp and n.

sp	n	Operation
any	0	All files assigned to the job at checkpoint time are processed according to the last operation performed.
0	≠ 0	All files assigned to the job at checkpoint time are processed. The n files included in the parameter list are processed according to their respective f values. All other files are processed according to the last operation performed.
≠ 0	≠ 0	The n files specified in the parameter list are processed according to their respective f values.

The procedure scratch files ZZZZZC0, ZZZZZC1, and ZZZZZC2 are always checkpointed; they are copied from BOI to EOI (refer to f_i =2), regardless of the sp and n values.

The INPUT, OUTPUT, PUNCH, PUNCHB, P8, and LGO files, and procedure data files ZZCCLAA through ZZCCLZZ are always checkpointed; they are copied according to the last operation performed (refer to f_i=3), regardless of the sp and n values.

For all other files except direct access files, the default copy type is f=4 when n≠0. For direct access files, the type of copy CHECKPT makes depends on the access mode.

Mode		User Option	Default
W	Write	Any type of copy	Copied (f=3)
R	Read-only	Any type of copy	Not copied (f=4)
E	Execute-only	Only f=4	Not copied (f=4)
A	Append-only	Any type of copy	Copied (f=3)
M	Modify	Any type of copy	Copied (f=3)
RA	Read/allow append	Any type of copy	Copied (f=3)
RM	Read/allow modify	Any type of copy	Copied (f=3)
U	Update	Any type of copy	Copied (f=3)
RU	Read/allow update	Any type of copy	Copied (f=3)

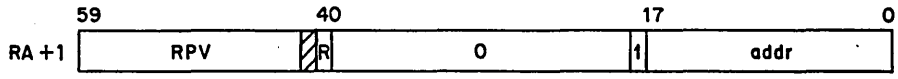
For a random file the copy type must be f=2 or it will be copied according to the last operation performed (f=3).

If f=4 is selected, you may have to retrieve the file yourself at restart time and select the NA and FC options on the RESTART command (refer to Volume 3, System Commands).

REPRIEVE PROCESSING

Reprieve processing (routine RPV) enables the user program to intercept interrupts prior to an abort of the program and to take corrective action. This action can be the completion of files (such as flushing buffers or closing files) and tables or it can be an attempt to correct the problem.

The format of the call to RPV is as follows:



R Auto recall bit.

addr First word address of the parameter block.

RPV supports the following functions.

- SETUP
- RESUME
- IRESUME
- RESET

The SETUP function is used to initialize the parameters for the RPV interface (that is setting the mask bits that indicate which errors are to be intercepted). It also initializes all RPV data in the job's control point area; if the parameter block indicates pending interrupts or I/O requests, they are processed at that time.

The interrupt handler is that portion of your program that processes interrupts. When an error occurs, the system will interrupt your program and restart it at the address that was specified in the RPV block. At this point, your program is running in interrupt handler mode which will prevent further errors from interrupting the program until the program has completed processing the first error.

The resume functions are used to restart the program after an interrupt has been processed. Any pending interrupts that occurred during the processing of the previous interrupt are detected then and the interrupt handler is restarted to process these interrupts. Optionally, the mask bits can also be changed during a resume function. Two modes of resume are allowed: program mode and interrupt handler mode.

NOTE

If the interrupt being processed is a CPU error exit, a resume function is not possible without manipulation of the exchange package and, in most cases, is not practical even with such manipulation.

The program mode resume (RESUME) will cause the system to restart the program at the interrupt point. The interrupt handler mode will be cleared so that any additional errors will cause a new interrupt of the program. Also, any errors which occurred during interrupt handler mode processing will cause a new interrupt at this time.

The interrupt handler mode resume (IRESUME) will cause the system to restart the program at the point of interrupt, but without clearing the interrupt handler mode. This allows the program to complete the task it was engaged in before the interrupt, and to then process the error and terminate the program at some later time.

NOTE

Further interrupts will be inhibited after the IRESUME is processed. Normal processing may be resumed by issuing another REPRIEVE SETUP.

The RESET function is used to reset a previous error flag and allow the operating system error handling routines to process the error. That is, the mask bits are cleared to disable further reprieve processing and the system processes the error as if the appropriate mask bit had not been set. Pending interrupts are not processed; however, any pending RA+1 request is reset and the exchange package is reset from the reprieve parameter block.

RPVBLK

The RPVBLK macro creates a REPRIEVE parameter block.

Macro format:†

Location	Operation	Variable
tag	RPVBLK	xfer,xjp,chklwa

tag Symbol for address of parameter block.

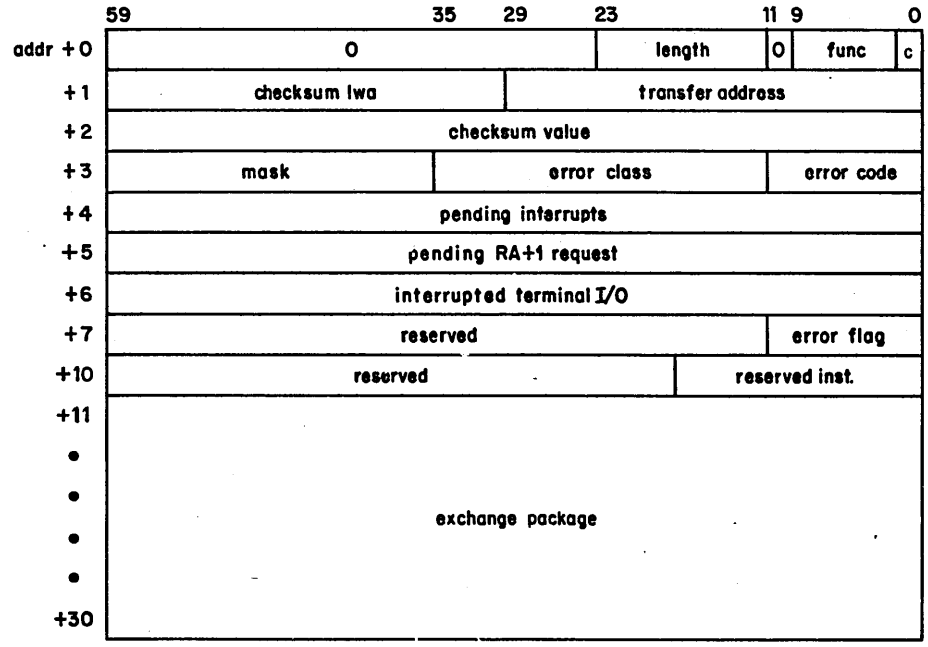
xfer Address of error processing routine (transfer address).

xjp Symbol for address of exchange package within parameter block (optional).

chklwa Checksum last word address (optional).

†This macro is not available in SYSTEXT. The user program must specify systems text PSSTEXT or call common deck COMCCMD (refer to appendix F). This macro requires common deck COMSRPV.

The format of the parameter block is as follows:



length	Length of the parameter block including the exchange package area (minimum of 31_8 words).
func	Function code: 1 SETUP. 2 Program mode resume (RESUME). 3 RESET. 4 Interrupt handler mode resume (IRESUME).
c	Completion bit (set by RPV when operation is complete).
checksum lwa	Specified by the user to indicate the end of the area to be checksummed and compared or set. If zero, no checksum is desired (checksum area begins at transfer address).
transfer address	Address to which control is transferred when an interrupt is processed.
checksum value	Either set to the checksum of the indicated area when RPV is called, or compared against the computed checksum (if checksum lwa is specified) when a reprievable error is processed.

mask

Indicates which types of interrupts are to be intercepted so that they may be processed in the program. If a particular bit is set in the mask, the corresponding error class will be intercepted; if a bit is clear in the mask, the corresponding error class will be processed by the system. If mask=0, extended reprieve processing is cleared.

mask	Error Class
001	CPU error exit.†
002	PP call error.
004	SRU limit.
010	Operator termination.††
020	PP abort.
040	CPU abort.
100	Normal termination.
200	Terminal interrupt.†††

error class††††

Set to the value of the mask bit which intercepts the indicated error (that is, if error x is intercepted by mask bit n, then bit n in the error class field is set). Refer to table 10-1 for a list of error classes.

error code††††

Octal code indicating error encountered. Refer to table 10-1 for a list of error codes and their meanings.

pending interrupts

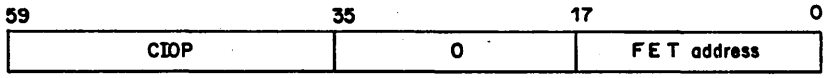
Used to queue pending interrupts (that is, the nth error code sets bit n in this field).

pending RA+1 request†††††

Contents of RA+1 at time of interrupt. RA+1 is reset from this field on a resume or reset call.

interrupted terminal I/O†††††

Contains interrupted input request if an interrupt occurs while a terminal input request is pending. The format is as follows:



error flag††††

Value of the operating system error flag at the time of the interrupt (refer to appendix E for a list of error flags).

---

†The MODE macro, described in section 6, can be used to specify which types of arithmetic operations will cause this error to be detected.

††NOS will allow a job to reprieve an operator KILL only once; the second time the operator KILLs the job, no reprieve processing will occur.

†††Refer to Control of Program Execution in section 12 for a description of the methods by which you can interrupt a program from a terminal.

††††Because the binary values of the error flags are subject to change in subsequent releases of NOS, you are encouraged to use the error class and error code fields for checking error status rather than the error flag field.

†††††If this field is nonzero, the FET involved has been left in a busy state. If the program wishes to use this FET within the interrupt handler (prior to doing a REPRIEVE RESUME), it is the program's responsibility to set the FET complete before trying to use it.

reserved inst.	This area is reserved for use by the installation.
exchange package	A copy of the exchange package at the time of the interrupt (unchanged from the executing package at the time of the error). This is the exchange package that is used when the interrupt handler is started.

Table 10-1 lists the return information for RPV error codes, classes, and flags.

Table 10-1. RPV Error Codes, Classes, and Flags

Error Code(s)	Description	Error Class	Corresponding Error Flags
0	Normal termination	100	None
1	Time limit	004	TLET
2	CPU error exit†	001	ARET, PEET, PSET
3	PP abort, SCP invalid transfer address	020	PPET, MLET, ITET
4	CPU abort	040	CPET
5	PP call error	002	PCET
6	Operator drop	010	ODET, IDET, STET
7	Operator kill	010	SSET, OKET, ORET, SYET, FSET
10	Operator rerun	010	RRET
11	Command error	040	None
12	Extended memory parity error	020	ECET
15	Auto recall error	002	None
16	Job hung in auto recall	002	RCET
17	Mass storage/service limits	004	FLET, TKET, SLET
20	PP program not in library	002	None
21	I/O limits	004	SRET
34-37	Reserved for installations	N/A	None
40	Terminal interrupt††	200	TIET, TAET
N/A	Reserved for installations	2000	None
N/A	Reserved for installations	4000	None
N/A	N/A	N/A	DRET, TJET, RAET

†The MODE macro, described in section 6, can be used to specify which types of arithmetic operations will cause this error to be detected.
†† Refer to Control of Program Execution in section 12 for a description of the methods by which you can interrupt a program from a terminal.

Refer to appendix E for a complete list of error flags and descriptions.

REPRIEVE

The REPRIEVE macro provides access to the extended reprieve processing capabilities of RPV. REPRIEVE parameters are inserted into the designated positions in the parameter block. Other fields are the responsibility of the caller.

Macro format:†

Location	Operation	Variable
	REPRIEVE	addr,type,mask

addr — Address of the parameter block.

type — Type of call: SETUP, RESET, RESUME (program mode), or IRESUME (interrupt handler mode).

mask — Desired mask setting.

The REPRIEVE macro issues the RA+1 request without using common deck COMCSYS, since that routine is not reentrant and the interrupt could have occurred there. The interrupt handler should not access any other portions of the program which are not reentrant or are shared with noninterrupt handler code.

One way to avoid this problem is to qualify those macro calls in the interrupt handler that use COMCSYS to issue the requests, and also provide a qualified copy of COMCSYS to ensure that no conflict exists. The following example illustrates this process with the MESSAGE macro.

```
            Interrupt handler code
             .
             .
             .
            QUAL    IHX
            MESSAGE ERR1,3
            QUAL    *
             .
             .
             .
ERR1        DATA    C*SOME ERROR*
            QUAL    IHX
QUAL$       SET     0   FORCE UNQUALIFIED COMMON DECKS
OPL         XTEXT   COMCSYS
            QUAL    *
             .
             .
             .
```

Appendix M contains an illustration of extended reprieve processing.

†This macro is not available in SYSTEXT. The user program must specify systems text PSSTEXT or call common deck COMCCMD (refer to appendix F).

SYSTEM REQUESTS

The following requests perform miscellaneous tasks associated with a job. Most of the requests are processed by the system monitor directly rather than by a specific function processor. The calling format is shown for each macro. Unless otherwise noted, the only common deck required for absolute assemblies is COMCSYS.

ABORT

A user program can abort itself when a program error occurs by using the ABORT macro. If the job or the current procedure contains an EXIT command, the system continues job processing with the command that immediately follows the EXIT command. This process can be altered either by reprieve processing (refer to section 10) or by using the NOEXIT command (refer to Volume 3, System Commands).

Macro format:

Location	Operation	Variable
	ABORT	

The format of the RA+1 call for this function is:

DAYFILE

The DAYFILE macro enables the user program to access the job dayfile. A FET of at least seven words must be specified.

A portion of the dayfile resides in the system central memory buffers. This is transferred to the buffer in the job's field length specified by the FET pointers FIRST and LIMIT. You must ensure that the buffer specified is large enough to accommodate the central memory portion of the dayfile. The released size is 114_8; if the value is different at your site, you should contact your site personnel to determine what it is.

That portion of the dayfile that resides on mass storage is made available to the user program as a library file assigned to the job in read-only mode. If a local file with the same name as the file name specified already exists, it is returned. The file is positioned at BOI.

This macro requires common decks COMCSFM and COMSSFM for both relocatable and absolute assemblies and also requires common deck COMCSYS for absolute assemblies.

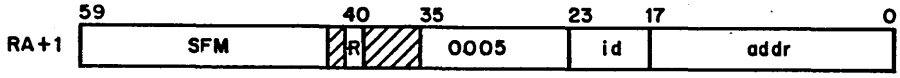
Macro format:†

Location	Operation	Variable
	DAYFILE	addr,type

addr — Address of the FET for the call; name given to the mass storage portion of the file.

type — USER (function 0005); attach dayfile. This is the dayfile of the job currently running.

The DAYFILE macro issues the following request to the function processor SFM.

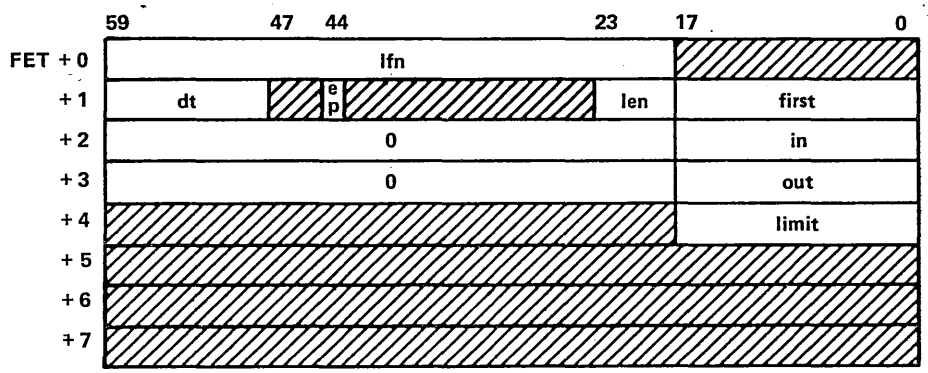


R — Auto recall bit (required).

id — File identification number.

addr — Address of the FET for the file.

The format of the FET used by SFM is:



†The macro is not available in SYSTEXT. The user program must specify system text PSSTEXT or call common deck COMCMAC (refer to appendix F).

lfn	Local file name of the file to which the job dayfile is to be returned.
dt	Device type of the device on which the local file resides (refer to Equipment Codes in appendix E).
ep	Error processing bit (refer to FET Description in section 2).
len	Specifies the length of the FET beyond five words.
first	First word address of input/output buffer (FIRST address).
in	The next available location for entering data into the buffer (IN address).
out	The next available location for removing data from the buffer (OUT address).
limit	Last word address of the buffer plus 1 (LIMIT address).

The IN and OUT FET pointers are updated after this request to indicate the presence of data in the buffer.

The following example reads the user dayfile and writes it to OUTPUT (assembly of this program requires that the system OPL be attached to the job).

```
          IDENT  DAYF
          ENTRY  DAYF
          SYSCOM B1
OPL       XTEXT  COMSSFM

**        ASSEMBLY CONSTANTS.

CBUFL     EQU    400B          CM BUFFER LENGTH
BUFL      EQU    301B          DAYFILE/OUTPUT BUFFER LENGTH
WBUFL     EQU    300B          WORKING BUFFER LENGTH

**        FETS AND CONSTANTS.

I         BSS    0             DAYFILE
SCR       FILEB  IBUF,BUFL,FET=7

O         BSS    0             OUTPUT
OUTPUT    FILEB  OBUF,BUFL,FET=7

WDCT      DATA   0             WORD COUNT OF CM BUFFER

**        MAIN PROGRAM.

DAYF      SB1    1
          DAYFILE I,USER

*         SAVE CM BUFFER PORTION OF DAYFILE.

          SA1    I+1           COMPUTE CM BUFFER WORD COUNT
          SA2    A1+1
          SX1    X1
          IX6    X2-X1
          ZR     X6,DAY1       IF CM BUFFER EMPTY
          SA6    WDCT
          READW  I,CBUF,X6     READ CM BUFFER

*         TRANSFER MASS STORAGE PORTION OF DAYFILE.

DAY1      READ   I             INITIATE FILE READ
DAY2      READW  I,WBUF,WBUFL
          NZ     X1,DAY3       IF TRANSFER COMPLETE
          WRITEW O,WBUF,WBUFL  CONTINUE TRANSFER

          EQ     DAY2

DAY3      SX2    B6-WBUF       SET WORDS TRANSFERRED
          ZR     X2,DAY4       IF NONE
          WRITEW O,WBUF,X2     EMPTY BUFFER

*         WRITE CM BUFFER PORTION OF DAYFILE.

DAY4      SA2    WDCT          GET CM BUFFER WORD COUNT
          ZR     X2,DAY5       IF CM BUFFER EMPTY
          WRITEW O,CBUF,X2     WRITE CM BUFFER

DAY5      WRITER O
          RETURN I
          ENDRUN

**        COMMON DECKS.

OPL       XTEXT  COMCSFM

**        BUFFERS.

IBUF      BSS    BUFL          DAYFILE BUFFER
OBUF      BSS    BUFL          OUTPUT BUFFER
CBUF      BSS    CBUFL         CM BUFFER
WBUF      BSS    WBUFL         WORKING BUFFER

          END    DAYF
```

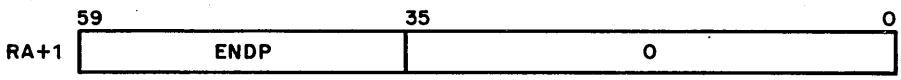
ENDRUN

The ENDRUN macro requests normal termination of a program. The system begins execution with the next command, if there is one. If there are no more commands or if the next command is an EXIT command, the system terminates the job. This process can be altered by reprieve processing (refer to section 10).

Macro format:

Location	Operation	Variable
	ENDRUN	

The format of the RA+1 call for this function is:



GETLIDA

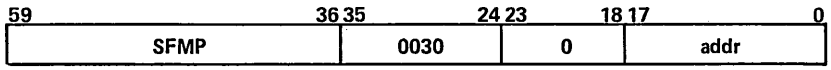
The GETLIDA macro allows the user program to obtain the various attributes for a specified LID. Common decks COMSSFM and COMCSFM are required for absolute and relocatable assemblies.

Macro format:†

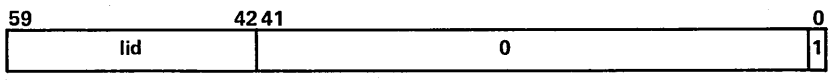
Location	Operation	Variable
	GETLIDA	addr

addr Address to receive attributes of specified LID.

The format for the RA+1 call for this function is:



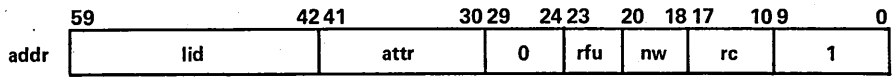
The format of addr word (before the call to COMCSFM) is:



lid User-specified lid.

†This macro is not available in SYSTEXT. The user program must specify systems text PSSTEXT or call common deck COMCCMD (refer to appendix F).

The attributes of the lid are returned in the following format:



lid User-specified lid.

attr lid attributes (zero if lid not found).

Bit(s)	Description
41	Set if host lid.
40	Set if linked lid.
39	Set if lid is enabled.
38	Set if prevalidation is required for lid.
37	Set if store and forward lid on host.
36	Set if store and forward lid on linked mainframe.
35	Set if loopback lid on host.
34-32	Reserved for CDC.
31-30	Reserved for installation.

rfu Reserved for CDC.

nw Network enabled flags.

Bit(s)	Description
20	Set if Scope 2 station is in service.
19	Set if Remote Host Facility is in service.
18	Set if Network Host Products is in service.

rc SFM response code (zero if normal termination).

GETMC

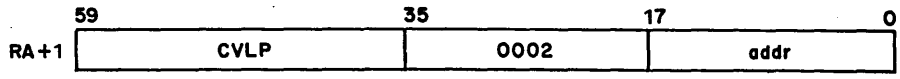
The GETMC macro allows the user program to obtain information about machine characteristics and the system environment. Common deck COMCCVL is required for absolute and relocatable assemblies.

Macro format:†

Location	Operation	Variable
	GETMC	addr

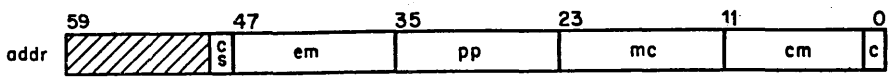
addr Address to receive machine characteristics.

The format of the RA+1 call for this function is:



†This macro is not available in SYSTEXT. The user program must specify systems text PSSTEXT or call common deck COMCMAC (refer to appendix F).

The machine characteristics are returned in the following format.



cs If bit 48 is set, system is operating in 64-character set mode.

If bit 48 is clear, system is operating in 63-character set mode.

em Extended memory size/1000_8.

pp Number of PPs in system.

mc Machine characteristics.

Bit(s)	Description
23-22	Unused.
21	Set if extended memory is in ESM mode.
20	Set if machine is a CYBER 180-class model or models 865 or 875.
19-18	Nonzero if CPU is a model 176; zero if CPU is not a model 176.
17	2x PP speed.
16	Set if machine is CYBER 170, except CYBER 180-class models.
15	Set if compare move unit (CMU) option is present.
14	Always set to indicate that CEJ/MEJ instruction is available.
13	Set if CPU0 has instruction stack.
12	Set if CPU1 is present.

cm Central memory size/200_8 (set to 3777_8 if central memory$\geq$262K).

c Completion bit (set when operation complete).

GETSSM

The GETSSM macro allows the user program to determine if the system is running in a secured mode.

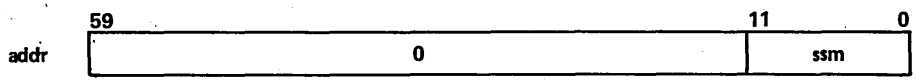
Macro format:†

Location	Operation	Variable
	GETSSM	addr

addr Address to receive system security mode.

†This macro is not available in SYSTEXT. The user program must specify systems text PSSTEXT or call common deck COMCMAC (refer to appendix F).

The system security mode is returned as follows:



ssm	Description
0	If System is in unsecured mode.
1	If System is in secured mode.

MEMORY

The MEMORY macro enables you to determine or change the amount of central memory or extended memory associated with your job or determine the maximum amount of memory you can request.

Macro format:

Location	Operation	Variable
	MEMORY	type,stat,R,fl,na

type — CM or null for central memory; either is legal. ECS for extended memory.

stat — Status word address. May be omitted only if the fl parameter is specified and na is null.

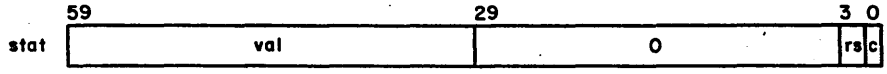
R — Auto recall bit.

fl — Desired new field length which, if specified, overrides the stat word. If fl is specified, the MEMORY macro sets the upper 30 bits of the stat word to the value of fl. A negative value may not be used for fl.

If the value of fl is greater than 377777_8, the value must be passed to the MEMORY macro in an X register.

na — If null, the program is aborted if the request exceeds the current maximum which you are allowed. If this parameter is specified, the program is not aborted if the request exceeds the current maximum. Instead, the macro does not change the field length, and sets the stat word as defined below.

The format of stat word (if used) is:



val — Prior to the macro call, val is used to specify the desired new field length (words parameter can override this value). If val (or fl) is +0, then the current field length for the specified type is returned in this field. If bit 47 of val is set and specified type is CM, and memory reduction is honored even if no reduce has been selected (that is, no reduce override is in effect).† If val is -1, then the current maximum field length for the specified type is returned; if user ECS is disabled or not present, a value of zero is returned in val, regardless of any validated extended memory field length. If val (or fl) is greater than zero for the macro call, it contains the actual value assigned upon return. If val is -0 and type is extended memory, the field length is set to 0.

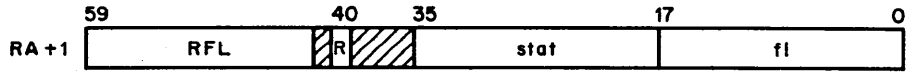
rs — These bits are reserved for system usage.

c — Completion bit. The system sets this bit when the request is complete.

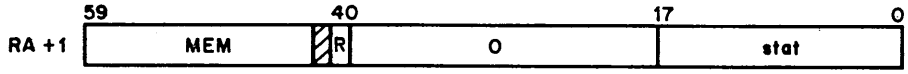
If a request is given for an amount greater than the current maximum field length (refer to the NOS 2 Installation Handbook for a description of the current maximum field length) and if na is specified, then no field length change occurs and control is returned with val set to the current field length value.

If the reserved bits (rs) in the status word are used, the MEMORY request or a subsequent MEMORY request may be aborted.

The format of the RA+1 call for this function is:



Alternate call format:††



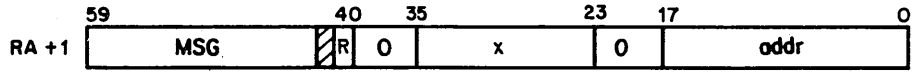
MESSAGE

The MESSAGE function allows the user program to display a message on the system console display and enter it in a dayfile.

If the job is of system origin, including a dollar sign as the first character of the message and routing the message with option 2 causes the message to be flashed on the B display.

The maximum length of a message is 80 characters; up to 40 characters per line are displayed. The message ends with either the first word containing 12 bits of zeros in any byte or at the 80th character. Before issuing the MESSAGE function, the user program must pack the display coded message in sequential locations, beginning at location addr.

The format of the RA+1 call for this function is:



†Refer to the REDUCE (-) command in the CYBER Loader Version 1 Reference Manual.
††To maintain compatibility with NOS/BE.

Macro format:

Location	Operation	Variable
	MESSAGE	addr,x,R

addr — Beginning address of the message. The beginning address of the message cannot be within five words of the end of the job's field length.

x — Message routing option.

x=0	Message is placed in the system dayfile, the user dayfile, and job message buffer 1.
x=1	Message is placed in job message buffer 1.
x=2	Message is placed in job message buffer 2.
x=3	Message is placed in the user dayfile and displayed at line 1 of the control point.
x=4	Message is placed in the user dayfile.
x=5	Message is placed in the user dayfile.
x=6	Message is placed in the system dayfile, the user dayfile, and job message buffer 1.
x=7	Message is placed in the user dayfile and job message buffer 1.†

If x is not specified, x=0 is assumed. If x is not defined or is an incorrect positive value, x=1 is assumed. If x is a negative number, unpredictable results will occur. If x is the character string LOCAL, x=3 is used.

The job message buffers (1 and 2) allow you to display concurrently messages that enter the dayfile and those that require operator action. Message buffer 1 is normally used to display information about the current status of the executing program; for example:

```
SKIPPING lfn

COPYING lfn

ASSEMBLING TEST
    .
    .
    .
```

Message buffer 2 is used by the system to issue error messages and warnings, and can be used by applications programs for the same purpose. Messages such as the following should be placed in message buffer 2:

```
BUFFER OVERFLOW, FILE lfn

FILE lfn NOT IN QUEUE

TIME LIMIT, REPRIEVE INITIATED
```

Message buffer 1 has a length of 50 characters (5 CM words); message buffer 2 has a length of 30 characters (3 CM words).

†Provided for compatibility with NOS/BE.

These message buffers are displayed at the system console and can be displayed by user jobs using the ENQUIRE command. If there are messages in both buffers, the message in message buffer 2 is displayed; the message in message buffer 1 is displayed only if message buffer 2 is empty (contains binary zero).

The contents of the job message buffer(s) are displayed when you enter the immediate job status command cE from an interactive terminal (refer to section 8 of Volume 3, System Commands), and when you enter the ENQUIRE,UJN or ENQUIRE,JSN=jsn command to determine the status of one of your jobs.

Only messages that are of interest to other than the job, such as the commands processed and compilers used, should be placed in the system dayfile (x=0). All messages of interest to the job, such as the path taken by the programs and the number of records copied, should be placed only in the user dayfile (x=3). All messages placed in the user dayfile (x=0 and x=3) are counted by the system. If the number of messages issued by the job exceeds the limit for which you are validated, the job aborts.

R Auto recall bit.

If the MSG RA+1 request is issued directly (not using the macro) and the upper 12 bits of the location specified by the message address are zero, then bits 47 through 30 of this location are assumed to contain the beginning address of the message. This indirect addressing method is not supported by the macro call and unpredictable results will occur if it is attempted.

MOVE

The MOVE macro moves a block of data from $addr_1$ to $addr_2$. This macro requires the common deck COMCMVE for absolute assemblies.

Macro format:

Location	Operation	Variable
	MOVE	count,$addr_1$,$addr_2$

count Number of words in the block to be moved.

$addr_1$ Address of the first word of the block to be moved.

$addr_2$ Address of the first word of the destination.

MOVE allows overlap in data moves; that is, $addr_2$ can be less than $addr_1$ plus count.

RECALL

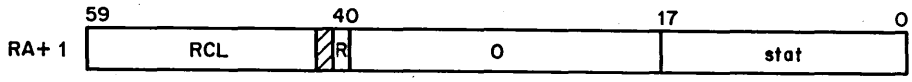
The RECALL macro enables the user program to relinquish the CPU until a function is completed or the CPU recall time has elapsed (delay time is set by the installation, usually 24 milliseconds). If the stat parameter is included in the call, control is not returned to the program until bit 0 of the word specified by stat is set. If stat is not included in the macro call, the program relinquishes the CPU until the delay time expires, or until a PP activity completes.

Macro format:

Location	Operation	Variable
	RECALL	stat

stat — If this parameter is present, control is returned to the program when bit 0 of the word specified by the address stat is set. This is usually the address of a FET.

The format of the RA+1 call for this function is:



The auto recall bit, R, is set only if the stat parameter is specified.

SUBR

The SUBR macro enables the user program to distinguish between entering a subroutine and exiting from a subroutine even though control is transferred to the same address. Transfers to the subroutine are of the form:

```
RJ tag
```

Exits from the subroutine are of the form:

```
EQ tagX
```

Macro format:†

Location	Operation	Variable
tag	SUBR	

SYSTEM

With the SYSTEM function, the user program can request the system to process any three-character request. Refer to Request Processors, section 1, for a list of request processors. A PP program can be called from a CPU program if the first character of the name is alphabetic. Documentation of these programs (refer to the DOCMENT command in Volume 3, System Commands) should be consulted for the functions available (for example, LFM, CPM). These should be used when macros do not exist to issue the functions desired.

†This macro is not available in SYSTEXT. The user program must specify systems text PSSTEXT or call common deck COMCMAC (refer to appendix F).

Macro format:

Location	Operation	Variable
	SYSTEM	req,R,p_1,p_2

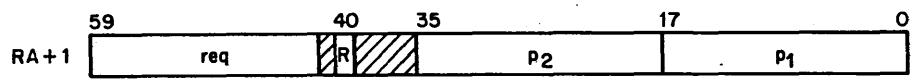
req — Three-character system request.

R — Sets the auto recall bit for the request. If specified, control is returned only after the request is completed. If = is specified, no executable code is generated, but cross-references are made.

p_1 — Bits 17 through 0 of the request.

p_2 — Bits 35 through 18 of the request.

The format of the RA+1 call is:



Example 1:

A CIO function request can be issued with the following SYSTEM request:

```
SYSTEM CIO,R,addr,n
```

where addr is the address of the FET for the file being read and n is the count for skip operations. If R is specified, the request is made with auto recall. When performing a CIO request in this manner, the user program must set the function code in FET+0.

Example 2:

The following shows the correct use of the = parameter:

```
SYSTEM XYZ,=
```

Routine XYZ will not actually be called, but the symbol XYZ will be created and KRONREF will recognize that this program calls XYZ.

WAIT

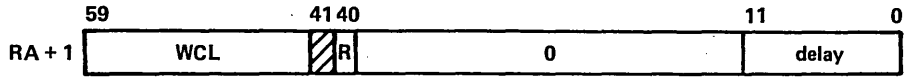
The WAIT macro enables the user program to relinquish the CPU and remain in recall until a specified CPU delay time has elapsed. The delay time can range from the system default recall time (10 milliseconds is the released default value) to 4095 milliseconds.

Macro format:

Location	Operation	Variable
	WAIT	delay

delay — The length of time in milliseconds that the job will remain in recall (if a value greater than 4095 is specified, the system uses the value 4095).

The format of the RA+1 call for this function is:



R — Auto recall bit.

USER FIELD LENGTH DUMP REQUESTS

The following requests enable you to transfer information residing in your job's field length to a file. Common deck COMCSYS is required for absolute assembly of these requests. These requests can not be made from a program in which the SETSSM macro has been executed, or from a program that was loaded from an execute-only file.

The SYSTEM macro is used to issue the user field length dump requests. Each type of request is identified by a three-character request code that appears as the first element in the variable field of the SYSTEM macro.

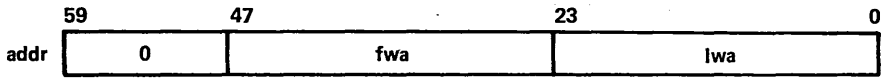
DUMP EXTENDED MEMORY (DEP)

This request dumps extended memory to file OUTPUT.† The format of the request is:

```
SYSTEM DEP,R,addr
```

R Auto recall bit.

addr The address of a parameter word with the following format.



fwa The address of the first word to be dumped. If this parameter is not specified, the dump begins with the first word of your job's extended memory.

lwa The address of the last word to be dumped. This parameter must be specified.

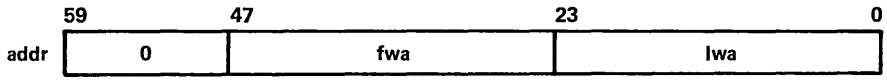
DUMP EXTENDED MEMORY WITH DISPLAY CODE (DED)

This request dumps extended memory to file OUTPUT. The dump contains display code equivalents to the right of the octal word representations. The format of the request is:

```
SYSTEM DED,R,addr
```

R Auto recall bit.

addr The address of a parameter word with the following format.



fwa The address of the first word to be dumped. If this parameter is not specified, the dump begins with the first word of your job's extended memory.

lwa The address of the last word to be dumped. This parameter must be specified.

†If the file OUTPUT is assigned to an interactive terminal, the dump is written to file ZZZDUMP instead of to OUTPUT.

BINARY DUMP OF CENTRAL MEMORY AND EXTENDED MEMORY (DMB)

This request creates a machine-readable binary dump file of your job's exchange package, central memory, and extended memory on file ZZZZDMB. File ZZZZDMB is not rewound before or after the dump; if ZZZZDMB is assigned to your terminal before the dump, the system returns it and creates a mass storage file ZZZZDMB. Each DMB request creates one logical record on ZZZZDMB. The format of the request is:

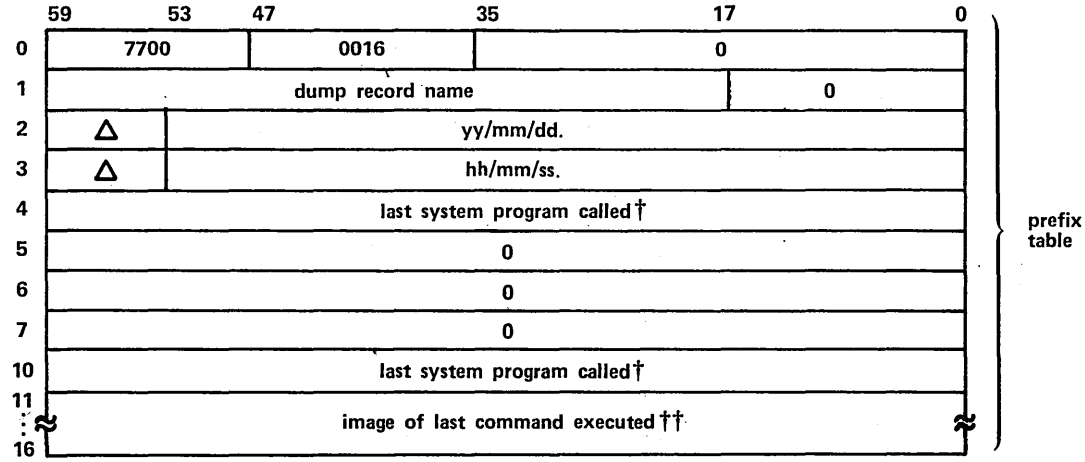
```
SYSTEM DMB,R,em,rn
```

R — Auto recall bit.

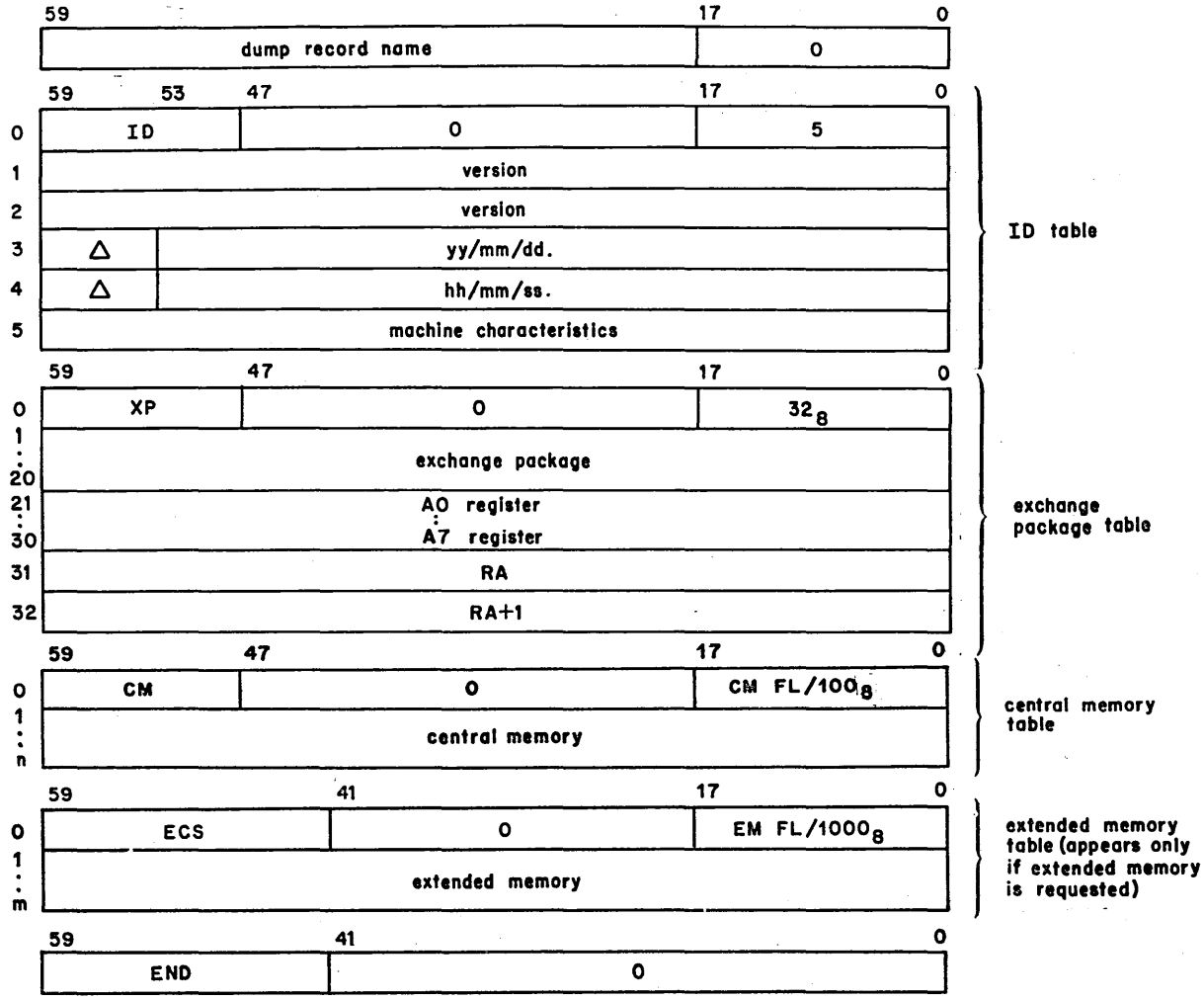
em — Determines whether the system dumps extended memory. If a nonzero integer is specified, the system dumps extended memory. If em is zero or omitted, the system dumps only the exchange package and central memory.

rn — Identifies the record name on file ZZZZDMB to which the dump will be written. rn is optional, but if it is specified it must be an octal value from 0 through 777777_8. The system appends rn to the letter D to form a seven-character dump record name. If rn is greater than 377777_8, the job aborts after the dump completes.

Each logical record of ZZZZDMB has the following format:



†Word PGNR of the job communication area.
††Words CCDR of the job communication area.

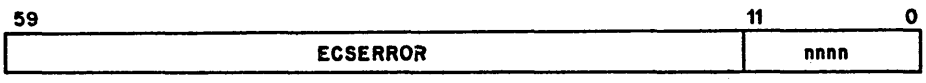


n Octal number of central memory words assigned to the job.

m Octal number of extended memory words assigned to the job.

Δ Space (display code value 55_8).

If an extended memory error occurs, the block in error will be filled with 400_8 words of the following format.



nnnn Word number of the block in error (starting with 1).

DUMP CENTRAL MEMORY (DMP)

This request dumps central memory of your job's exchange package to file OUTPUT.† The format of the request is as follows:

SYSTEM DMP,R,lwa,fwa

R	Auto recall bit.
lwa	The address of the last word to be dumped. This parameter must be specified if fwa is specified.
fwa	The address of the first word to be dumped. If this parameter is not specified but lwa is specified, the dump begins with the first word of your job's central memory.

If neither fwa nor lwa is specified, the dump includes the exchange package and 40_8 central memory locations before and after the program address in the exchange package. If either fwa or lwa is nonnumeric, the request is processed as if neither parameter had been specified.

DUMP CENTRAL MEMORY WITH DISPLAY CODE (DMD)

This request dumps your job's central memory or exchange package to file OUTPUT.† The dump contains display code equivalents to the right of the octal word representations. The format of the request is as follows:

SYSTEM DMD,R,lwa,fwa

R	Auto recall bit.
lwa	The address of the last word to be dumped. This parameter must be specified if fwa is specified.
fwa	The address of the first word to be dumped. If this parameter is not specified but lwa is specified, the dump begins with the first word of your job's central memory.

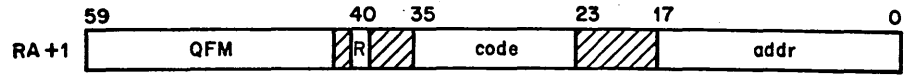
If neither fwa nor lwa is specified, the dump includes the exchange package and 40_8 central memory locations before and after the program address in the exchange package. If either fwa or lwa is nonnumeric, the request is processed as if neither parameter had been specified.

QUEUE FILE MANAGER

Queue file manager (QFM) performs functions associated with queue and dayfile protection. The majority of the QFM functions are for special system jobs; these functions are not discussed in this manual (refer to the NOS 2 Systems Programmer's Instant) since they are not available to the applications programmer. Functions available to the applications user allow a job to be rerun in the event of a system interrupt or a hardware failure.

†If the file OUTPUT is assigned to the terminal, the dump is written to file ZZZZDMP instead of to OUTPUT.

The format of the call to QFM is:



R	Auto recall bit.
code	Function code.
addr	Address of the request.

RERUN (015)

The RERUN macro sets the job rerun status, indicating that the job may be rerun in the event of system failure. All noninteractive jobs are initially given rerun status. The RERUN macro has no effect if used from an interactive job.

Macro format:†

Location	Operation	Variable
	RERUN	

NORERUN (016)

The NORERUN macro clears the job rerun status (initially enabled) and prevents a job from being rerun. With this macro the user program can specify when a job may be safely rerun. For example, if a job has just updated a critical data base, it is probably desirable not to rerun the job. The user program can use the NORERUN macro to prevent this. This macro affects noninteractive jobs only. The NORERUN macro has no effect if used from an interactive job.

Macro format:†

Location	Operation	Variable
	NORERUN	

LOADER REQUESTS

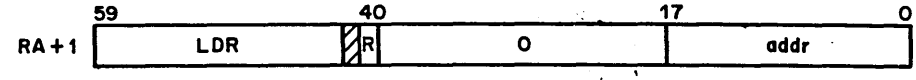
The system provides routines to aid you in loading overlays or capsules at specific points during program execution. The overlays or capsules can reside on files attached to your job or in system libraries.

†This macro is not available in SYSTEXT. The user program must specify systems text PSSTEXT or call common deck COMCMAC (refer to appendix F).

OVERLAY

The OVERLAY macro processes a system request to the LDR processor. LDR provides the ability to load overlays to specified areas of the user program area. Depending on the parameters specified and the level of the overlay, control may or may not be returned to the calling program.

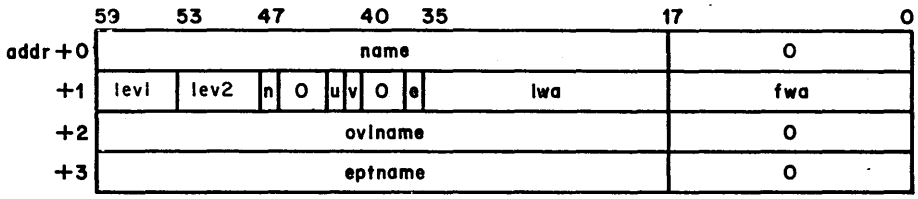
The format of the RA+1 call for this function is:†



R Auto recall bit.

addr Address of the request.

The load request consists of two to four words. The two- to four-word block must be defined by the user program for RA+1 calls, but is defined by the system when called from the OVERLAY macro (only two words used).



name Source of name depending on the u and n options.

levl First overlay level.

lev2 Second overlay level.

n Number of words in request-2 (bits 47-46).

u Load option (bit 42).

v Overlay flag (must be set to 1) (bit 41).

e Call completion flag (bit 36).

lwa Last word address available for load.

fwa The address at which the overlay is to be loaded.

ovlname Name of overlay to be loaded (if n≥1 and u=1).

eptname Entry point name when loading multiple entry point overlay (if n=2 and u=1).

---

†LDV is processed by the system as the LDR call. This is to provide compatibility with the common product set.

The operation performed depends on the value of u, n, fwa, lev1, and lev2.

1. If u=0, n is ignored and name is the name of the file containing the overlay (lev1 and lev2 are required).

2. If u=1 and n=0, name is the name of an entry point in an overlay from the system library† (lev1 and lev2 are ignored).

3. If u=1 and n≠0, ovlname is the name of the overlay from the system library† (lev1 and lev2 are ignored).

4. If u=1 and n=2, eptname is the name of an entry point in an overlay from the system library† (lev1 and lev2 are ignored).

5. If fwa=0, the overlay is loaded at the address specified by the overlay. Otherwise the overlay is loaded at fwa.

6. If lev1=lev2=0, the (0, 0 overlay) control is transferred to the called overlay; otherwise, control is returned to the caller with fwa replaced by the entry point address of the overlay.

7. If e=1, control transfers to the specified entry point, eptname, in the overlay.

Upon completion of the load, the following information is returned to the four-word block beginning at location addr:

	59–54	53–48	47–18	17–0
addr+0	name			0
+1	lev1	lev2	0	eptaddr
+2	ovlname			0
+3	eptname			0

eptaddr Entry point address of the overlay; if n=2, eptaddr is the address of eptname.

Macro format:

Location	Operation	Variable
	OVERLAY	nam,lev,SYSTEM,fwa

nam Address of file name in L format (display code, left-justified).

lev Level of overlay. If not specified, level 0,0 is assumed and control is automatically transferred to transfer address encountered on overlay load. (Usually specified on IDENT pseudoinstruction of ABS programs.) For overlay level (i,j), level is defined as: lev=i*100$_8$+j.

SYSTEM If SYSTEM is specified, file is loaded from system library and nam is name of overlay desired.

fwa If this parameter is specified, fwa is the address where the overlay is to be loaded. The file is loaded at the address specified on the overlay if this parameter is not specified.

Common decks required are COMCOVL and COMCSYS.

†The overlay is loaded from a user global library set file if the main program was loaded from a user global library set file.

After the macro is processed, X1 contains the address of the entry in the overlay.

Example 1:

```
                    .
                    .
                    .
            OVERLAY TEST
                    .
                    .
TEST        CON     4LTEST
```

This sequence of code loads overlay (0,0) from file name TEST and begins execution at the entry address specified by TEST.

Example 2:

```
            OVERLAY A,0100B,S,LDA
            SB2     X1
            JP      B2

                .
                .
A           CON     5LARITH
                .
                .
LDA         EQU     *
```

This sequence of code loads the ARITH routine from the system, level (1,0) library at address LDA and transfers control to the routine.

If the request calls for the overlay to be loaded and executed and no fwa load address is specified in the request, the entry point address specified in the load table is used as the entry point address. If an fwa load address is specified in the request, bias the entry point address from the load table with the difference of the fwa load address specified by the request and the fwa load address specified by the load table; that is:

Entry point address = (entry point address from load table) + (fwa load address specified in the request) - (fwa load address from load table)

If the request requires just an overlay load, the entry point address returned is the entry point address in the load table.

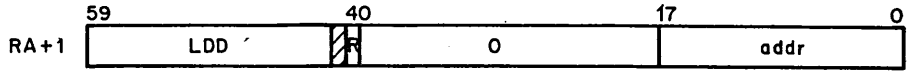
NOTE

Overlay loads from an execute-only file are always allowed in the case of a (0,0) overlay. For any other overlay, a load from an execute-only file is only allowed if the original (0,0) overlay was loaded from the same file.

LOADD

The LOADD macro allows the user program to locate procedures or fast dynamic load (FDL) capsules and have pertinent information returned to a specified address. Fast dynamic loading is a method of loading preprocessed binary routines.

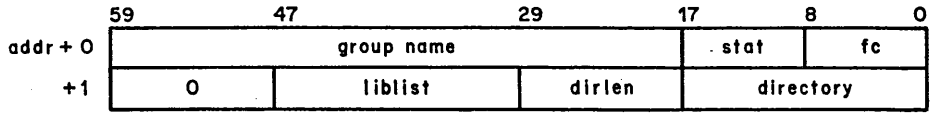
The format of the RA+1 call for this function is:



R Auto recall bit.

addr Address of parameter block.

The parameter block consists of two words in the following format.



group name Name of the group of capsules or procedures for which a directory is requested.

stat Status of call (ignored during request). Upon completion of call, stat is set to one of the following values.

Value	Description
0	Function completion without error.
1	Illegal function code.
2	Bad directory address or length (address plus length must be less than or equal to field length).
3	Bad liblist address or length.

The following errors may be combined with those preceding:

10_8	An entry in the library list did not correspond to any known local or system library name, or an entry specified the name of file which was not a mass storage library.
20_8	The specified directory space was not large enough to contain the entire directory.

fc Function code. LDD sets bit 0 to one when the request is complete:

0	Specifies capsule (record type CAP).
404_8	Specifies procedure (record type PROC).

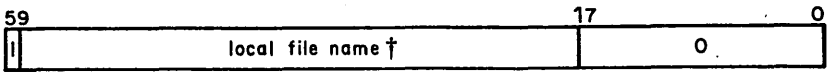
liblist Address of a list of libraries to be searched after the global library set. Zero if no set is specified.

dirlen Length, in central memory words, of the area to receive the generated directory. Upon completion of the call, it is set to the actual length that was needed for the complete directory (may be less than or equal to the value of the original call).

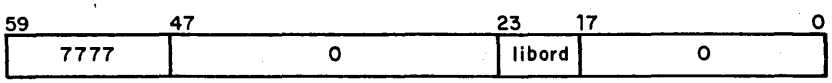
directory Address of the area to receive the generated directory.

When called, LDD searches first the global library set (refer to the CYBER Loader Version 1 Reference Manual for a description of global library sets) and then the library set specified in the call. If a library file is found to contain one or more capsules belonging to the given group or the specified procedure, an entry is made in the directory. This entry is one of two different forms, depending on whether the library is a system library or a local file library. For a procedure, LDD searches the central library directory (CLD) if the specified procedure is not found elsewhere.

For a local file library the format is:

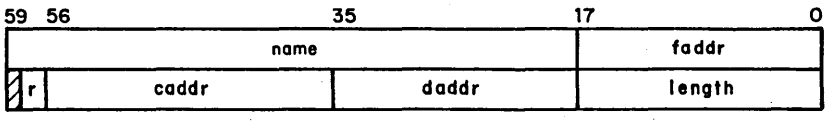


For a system library the format is:



libord The library ordinal of the library containing the capsule.

For each capsule or procedure found that belongs to the given group, LDD makes the following entry in the directory.



faddr Address, relative to the beginning of the directory, of the word containing the file entry associated with this capsule or procedure.

r Residence of capsule or procedure:

0	Mass storage.
1	Mass storage and CM.
2	Mass storage and extended memory.

† The first character of the local file name must not be numeric.

caddr — CM or extended memory address of capsule or procedure.

daddr — Disk address (relative PRU) of capsule or procedure.

length — Length of the capsule or procedure, including header, code image, and relocation and linking information, but excluding the prefix table.

Macro format:†

Location	Operation	Variable
	LOADD	addr,R

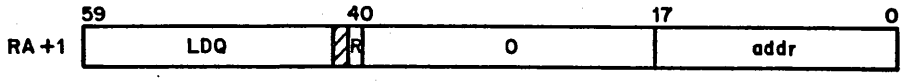
addr — Address of parameter block.

R — Auto recall bit.

LOADQ

The LOADQ macro loads fast dynamic load (FDL) capsules from specified files.

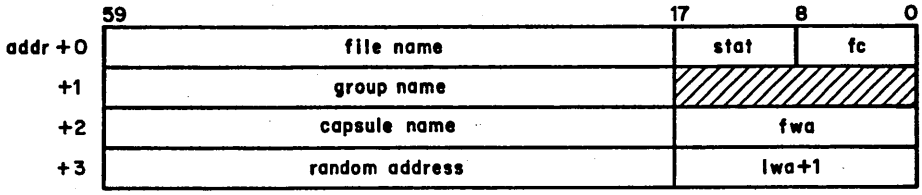
The format of the RA+1 call for this function is:



R — Auto recall bit.

addr — Address of parameter block.

The four-word parameter block must be defined as follows:



†This macro is not available in SYSTEXT. The user program must specify systems text PSSTEXT or call common deck COMCMAC (refer to appendix F).

file name — Name of file containing capsule.

stat — Status of LDQ call (ignored during request). Upon completion of call, stat is set to one of the following values.

Value	Description
0	Function completion without error.
1	Illegal function code.
2	Bad address (must have fwa<lwa+1<field length).
3	Nonexistent file or file not on mass storage.
4	Bad disk address (out of file bounds).
5	Capsule not found at specified location.
6	Insufficient space provided for capsule.

If either error 5 or 6 occurs, the contents of the loadable area are undefined.

fc — Function code:

0 Load capsule.

LDQ sets bit zero to one when the request is complete.

group name — Name of capsule group.

capsule name — Name of desired capsule.

fwa — First word address of the area into which the capsule is to be read.

random address — Location of capsule on specified file.

lwa+1 — Last word address plus 1 of area for capsule.

LDQ reads a capsule from the specified mass storage location, removing the prefix table, but not altering the record otherwise. LDQ ensures that the location contains a capsule (60 table). LDQ also determines that the entire capsule fits into the specified area and that the name is correct.

Macro format: †

Location	Operation	Variable
	LOADQ	addr,R

addr — Address of parameter block.

R — Auto recall.

MEMORY ALLOCATION FOR OVERLAY LOADERS

The overlay loaders load records into central memory. If a record begins with an overlay table (50_8-, 51_8-, and 53_8-identification word), it is loaded according to the relative address given in the table (figure 11-1). The beginning load address must be greater than 110_8. For a 50_8 single entry point overlay, loading starts at origin minus one; for a 51_8 multiple entry point overlay, loading starts at origin minus wc minus one, where wc is the number of entry points.

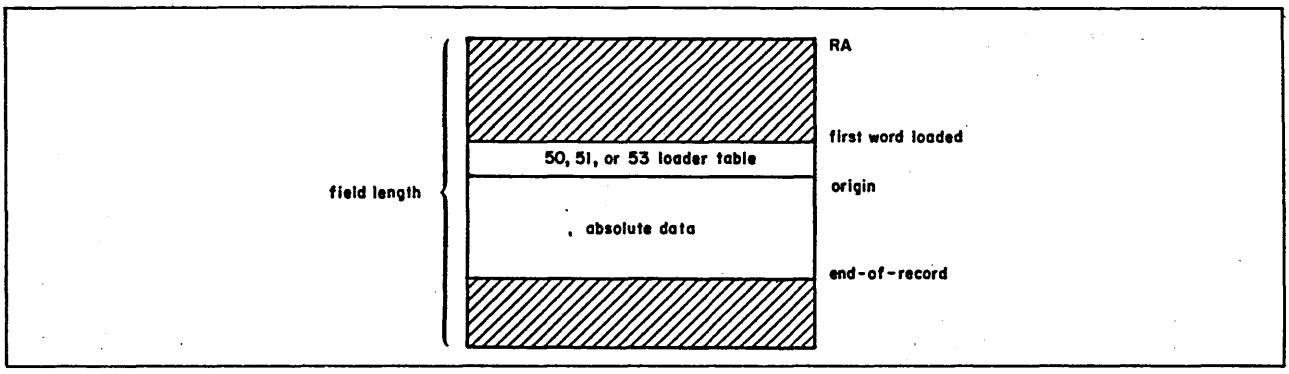


Figure 11-1. Absolute Loader Memory Assignment

† This macro is not available in SYSTEXT. The user program must specify systems text PSSTEXT or call common deck COMCMAC (refer to appendix F).

INTERACTIVE TERMINAL REQUESTS

The following requests perform functions associated with control of your interactive terminal. The calling format is shown for each macro.

NOTE

The macros described in this section are intended for use by interactive origin jobs only. The use of these macros by other job origins will be ignored.

DISTC MACRO †

The DISTC macro enables an interactive program to prevent IAF from processing terminal control keys and commands.

Capabilities similar to those provided by DISTC are available via reprieve processing (refer to Reprieve Processing, section 10). Because reprieve processing provides extended features, it should be used instead of DISTC.

Macro format:††

Location	Operation	Variable
	DISTC	st,addr,INT

st — Control status:

ON	Activates program terminal control.
OFF	Deactivates program terminal control.

addr — Specifies an interrupt address which the system uses to inform the program that a terminal operator attempted to interrupt or terminate the program. This parameter is valid only if st=ON. If addr is not specified, the program is not notified of attempted terminal control.

INT — If INT is specified when terminal control is attempted, the system copies the program operating registers (exchange package)††† to the 20_8-word area starting at addr. The program address is changed to addr+20_8 (that is, execution continues at addr+20_8). If INT is not specified, the contents of addr are determined by the type of user break attempted: if user break 1, the contents of addr are set to 1. If user break 2, the contents of addr are set to 2. The program can check the location periodically to determine if the terminal operator has attempted to interrupt or terminate the program.

---

†This macro will be deleted in a future version of NOS. Extended reprieve processing (refer to the REPRIEVE macro in section 10) should be used instead of DISTC.

††This macro is not available in SYSTEXT. The user program must specify systems text PSSTEXT or call common deck COMCMAC (refer to appendix F).

†††A system request, XJR, allows the user program to restore the operating registers and resume normal processing after the interrupt processing has been completed.

The interrupt address remains the same after the program is notified of an attempted interrupt. Once a terminal control is set or cleared, that status remains in effect until:

- The program issues a DISTC macro that changes the status.
- The program terminates, which returns the terminal to command mode.
- You log off or disconnect the terminal. If the terminal was disconnected, you may recover. If the executing program is continued after recovery, the terminal control status remains as it was prior to the disconnect. If the executing program is not continued, the interrupt address is cleared.

The following is a list of terminal control sequences and their effect when you select the disable terminal control feature (the (CR) denotes the message terminator key).

Phase	Sequence†	Effect
Waiting for input	CTRL/T (CR) or CTRL/P (CR)	You may enter this sequence at the beginning of any input line unless transparent input mode has been selected. If an interrupt address was specified, the program is notified of attempted control.
Generating output	BREAK CTRL/T (CR) or BREAK CTRL/P (CR)	If an interrupt address was specified, the program is notified of attempted control. When the BREAK key is pressed, output is suspended. However, unless CTRL/T (CR) or CTRL/P (CR) is entered next, the program is not notified of an attempted terminal control. If any other sequence is entered, or if just a (CR) is entered, the program is not notified of terminal control and the output operation continues.
Executing	CTRL/T (CR) or CTRL/P (CR)	If an interrupt address was specified, the program is notified of attempted control.

NOTE

If either REPRIEVE or EREXIT are present in the same routine with a DISTC call, they will take precedence over the DISTC call, and the DISTC will have no effect.

†The termination (sometimes referred to as user break 2) sequence (that is, CTRL/T(CR)) and interruption (sometimes referred to as user break 1) sequence (that is, CTRL/P(CR)) may vary for different terminals. Refer to Volume 3, System Commands, for further information.

CSET MACRO

The CSET macro sets the initial and current character set mode of the terminal to either ASCII or NORMAL.

Macro format:†

Location	Operation	Variable
	CSET	mode,C

mode	ASCII	Set ASCII 128-character set mode.
	NORMAL	Set ASCII graphic 63/64-character set mode.
	RESTORE	Set current terminal character mode to initial terminal character mode.
C		Change initial and current terminal character mode to that specified; if C is omitted, only the current mode is changed.

NOTE

The use of this macro may cause the character mode of the terminal to switch prior to the printing of all previous output. This can be prevented by preceding the macro call with an input request with recall or with a rollout request, or by flushing the output buffer with a WRITER or WRITEF request.

PROMPT MACRO

The PROMPT macro enables or disables the issuing of the terminal input prompt (a question mark). The input prompt is turned on at the beginning of each job step.

Macro format:†

Location	Operation	Variable
	PROMPT	mode

mode	Input prompt mode desired:	
	ON	Terminal input is requested with a question mark.
	OFF	No terminal input prompt is issued.

†This macro is not available in SYSTEXT. The user program must specify systems text PSSTEXT or call common deck COMCMAC (refer to appendix F).

SETSLM MACRO

The SETSLM macro returns or sets the display mode of the interactive terminal associated with a job.

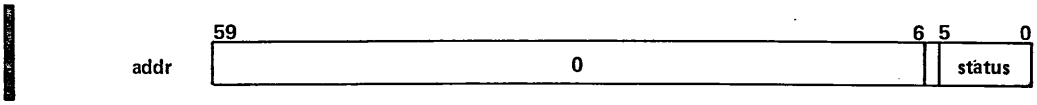
Macro format:†

Location	Operation	Variable
	SETSLM	addr,W

addr — Address of parameter word.

W — If present, the terminal's display mode is set according to the values specified in the parameter word. If omitted, the terminal's current display mode is returned to the parameter word.

Parameter word format:



status — Terminal display mode status.

Bit(s)	Description
0-5	Terminal model ordinal.

Ordinal	Mnemonic††	Terminal Type
0	None	Any terminal that does not support screen mode.
1	N/A	Reserved for CDC.
2	721	CDC 721.
3	722	CDC 722.
4	VT100	DEC††† VT100.
5	Z19	Zenith Z19††† or Heathkit H19.†††

Bit(s)	Description
6	Terminal display mode.

Value	Description
1	Screen mode.
0	Line mode.

†This macro is not available in SYSTEXT. The user program must specify systems text PSSTEXT or call common deck COMCMAC.

††These mnemonics and their corresponding ordinals are defined in common deck COMCGTO.

†††DEC is a registered trademark of Digital Equipment Corporation. The Zenith Z19 is a product of Zenith Radio Corporation. The Heathkit H19 is a product of the Heath Company.

TSTATUS MACRO

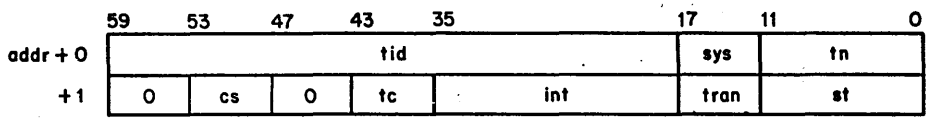
The TSTATUS macro returns the status of the terminal. Information returned includes terminal type, subsystem being used, the terminal number, parity, character set information, duplex and tape mode status, and the current interrupt address.

Macro format:†

Location	Operation	Variable
	TSTATUS	addr

addr Address of two-word status block.

Status block format:



tid Left-justified, blank-filled display code terminal type identification:

Type	Description
NAMIAF	Interactive terminal (only terminal type returned).

sys Subsystem ordinal:

Ordinal	Description
0	Null subsystem.
1	BASIC subsystem.
2	FORTRAN subsystem.
3	FTNTS subsystem.
4	Execute subsystem.
5	Batch subsystem.
6	Access subsystem.

tn Terminal number (octal).

cs Connection status.

Value (octal)	Description
0	Not interactive.
1	Detached.
2	Online.
3-13	Reserved for CDC.
14-17	Reserved for installation.

†This macro is not available in SYSTEXT. The user program must specify systems text PSSTEXT or call common deck COMCMAC (refer to appendix F).

tc Terminal class:

Class (octal)	Description
1	M33, M35, M37, or M38 TTY
2	CDC 713-10, 722, 751-1, 752, 756
3	CDC 721
4	IBM† 2741
5	M40 TTY
6	Hazeltine 2000
7	DEC† VT100
10	Tektronix 4010, 4014
11	HASP Protocol (postprint format control)
12	200 User Terminal
13	CDC 714-30
14	CDC 711-10
15	CDC 714-10/20
16	HASP Protocol (preprint format control)
17	CDC 731-12, 732-12
20	IBM 2780
21	IBM 3780
22	IBM 3270

int Current interrupt address if specified on DISTC macro; if INT is also specified on the DISTC macro, bit 35 is set.

tran Transmission code:

Code	Description
3	Interactive terminal (only code returned).

st Terminal status bits:

Bit	Description
0	Reserved.
1	Initial character set (ASCII if set).
2	Current character set (ASCII if set).
3	Reserved.
4	Reserved.

†DEC is a registered trademark of the Digital Equipment Corporation. IBM is a registered trademark of International Business Machines.

EFFECT MACRO

The EFFECT macro allows the user program to control the formatting of terminal output. Terminal output is normally controlled by IAF; when the mode is set to OFF, the user program must supply all formatting for terminal output.

Control bytes 0001, 0002, 0003, 0013, 0014, and 0015 should not be used while the mode is set to OFF, since use of these bytes may result in the loss of control over the carriage. This macro effects only terminals connected to the system by way of NAM.

Macro format:

Location	Operation	Variable
	EFFECT	mode

mode	ON	IAF controls all formatting of terminal output.
	OFF	USER program controls all formatting of terminal output.
	omitted	IAF controls all formatting of terminal output.

NOTE

For a list of format effectors and a discussion of the effect on a specific terminal class, refer to the NAM Version 1/CCP Version 3 Terminal Interface Reference Manual.

PROGRAM WRITING TECHNIQUES 12

WRITING PROGRAMS UNDER NOS

Listings of programs that use the macros and common decks described in this manual can serve as an aid to understanding many of the features described in this manual. For example, to obtain a listing of the COPYC program, enter the following commands after accessing the system OPL.†

```
MODIFY,LO=E,Z./*EDIT,COPYC
COMPASS,I,L=list,S=NOSTEXT,B=0.
ROUTE,list,DC=LP.
```

If you follow the documentation standards described in appendix I, you can use the documentation utilities to document your program. These specifications are supplied as a guide to enable you to understand the format of system programs and to give you an idea of how to structure programs.

You should organize the program area, buffers, and working storage areas in a manner that uses the least amount of memory for the shortest period of time. This provides better performance and more efficient use of system resources. Areas of code that are used once, such as preset routines, should be overlayed with buffers that are used later in the program. All main programs should be set to load past location 110_8 since areas before this address contain program and system parameters.†† If a program has several entry points, the origin should be higher. Placing FETs, table pointers, and other critical data areas at the beginning of the program aids in debugging because these areas do not change addresses as code in the program is changed.

PARAMETER PROCESSING

When a program is initiated, the command parameters are stored in the job communication area (refer to figure E-1) beginning at ARGR (RA+2) through RA+n (n cannot exceed 63_8). These parameters are terminated by a zero word. The number of parameters stored in the job communication area is placed in the lower 18 bits of ACTR (RA+64_8). The name of the command is placed in bits 59 through 18 of PGNR (RA+64_8). The parameters are stored in either operating system or product set format. The command image, less any label or prefix field, is stored beginning at CCDR (RA+70_8).

If the parameters are stored in operating system format, the comma, period, and right parenthesis are stored as zero. For all parameters and valid separators except the comma, their display code equivalent is stored.

---

† Your site must control access to the system OPL as required by the site's licensing agreement. Contact your site analyst to obtain the procedure for gaining access to the system OPL.

†† Refer to appendix E.

If the parameters are stored in product set format, separators and terminators are stored as follows:

Character	Code (Octal)
,	1
=	2
/	3
(	4
+	5
-	6
;	10
) or .	17
Others	16

Parameters for user programs are normally stored in product set format. You can force operating system format in either of two ways:

- Prefix the command with a slash (/).
- For programs loaded from a global library set, include an NPC= entry point in the program.

Example:

When a program is loaded from a local file with the following command:

```
TEST,A=B,C=D,E,F.
```

the command parameters are stored in product set format and the contents of central memory are as follows:

```
ARGR    RA+2    0100  0000  0000  0000  0002  A      B
          +3    0200  0000  0000  0000  0001  B      A
          +4    0300  0000  0000  0000  0002  C      B
          +5    0400  0000  0000  0000  0001  D      A
          +6    0500  0000  0000  0000  0001  E      A
          +7    0600  0000  0000  0000  0017  F      0
          +10   0000  0000  0000  0000  0000
          .
          .
          .
CCDR    RA+70   2405  2324  5601  5402  5603  TEST,A=B,C
        RA+71   5404  5605  5606  5700  0000  =D,E,F.
```

If the command is changed to:

```
/TEST,A=B,C=D,E,F.
```

the command parameters are stored in operating system format and the contents of locations RA+2 through RA+10_8 appear as follows:

ARGR	RA+2	0100	0000	0000	0000	0054	A	=
	+3	0200	0000	0000	0000	0000	B	
	+4	0300	0000	0000	0000	0054	C	=
	+5	0400	0000	0000	0000	0000	D	
	+6	0500	0000	0000	0000	0000	E	
	+7	0600	0000	0000	0000	0000	F	
	+10	0000	0000	0000	0000	0000		

NOTE

If you are using the common deck COMCARG (refer to appendix F) for argument processing, you do not need to be concerned with the different formats, since COMCARG processes both formats correctly.

WRITING INTERACTIVE PROGRAMS

You may choose to write interactive programs in languages other than those oriented specifically to the terminal (for example, COMPASS). You can also convert some batch application programs to interactive programs. The following description of system conventions acquaints the user with potential problem areas in performing this conversion or in writing interactive programs.

CONVERSION PROBLEMS

The major difference between an interactive job and a batch job is in the handling of files assigned to a terminal. Since the system is designed with a common control interface to all types of equipment, the applications programmer need not be concerned about the specific piece of equipment with which the program is communicating.

Certain problems arise when application programs not written with terminal interaction in mind are converted to interactive programs. These problems are:

- Each line of terminal input is considered a logical record. This causes two basic problems.
  - First, any program that terminates input processing on an EOR processes only one line of input.
  - Second, programs that read ahead on input place a special significance on an EOR, and therefore cannot effectively interact while accepting these one-line records. This problem is minimized by the design of NOS input/output. However, when the problem does occur, it is fatal to program execution.
- Many programs output more than 72 characters per line. This may result in unacceptable output; however, this is easily corrected by using the LO72 command or by reformatting the output in the program.
- Many programs perform unnecessary write requests on output. This can cause high system overhead.

Portions of the NOS standard product set are modified to correct these problems.

DEFAULT FILE ASSIGNMENTS AND SPECIAL FILE TREATMENT

All jobs interacting with your terminal have the INPUT and OUTPUT files assigned to the terminal unless you or a program assigns them to a different device. You can assign a file to the terminal by assigning a file to equipment TT using the ASSIGN command; you can assign the INPUT or OUTPUT file to mass storage by specifying a device type of MS on the ASSIGN command. A user program can make similar assignments with the REQUEST macro.

Normally, an interactive program need not check for EOI on INPUT; however, you must ensure that INPUT is checked for EOI file status if INPUT might be assigned to mass storage.

SPECIAL HANDLING

When a program makes a read request with recall on a terminal input file, the circular buffer of a terminal file can be automatically flushed (as if an EOR write had been performed) before the system issues the read request.

A program can use one of the following methods to ensure that output data is sent to the terminal in this way.

- The program can specify a list of files (refer to the SETLOF macro, section 6) from which the system determines the appropriate file name. This is the preferred method.
- The program can store FET pointers, which specify the files to be flushed, in RA+2 through RA+63$_8$. The FET pointer is the file name left-justified with zero-fill in bits 59 through 18 and the address of the FET in bits 17 through 0.

In using the first method, the system checks only the first file in the list. This file must meet the following criteria.

- The file must either be assigned to the terminal (device type equals TT in the FET) or, if unassigned, must be named OUTPUT.
- The file must meet the conditions described under the flush bit (refer to description of the flush bit, FET description, in section 2).

In using the second method, the system uses the first file that meets the preceding criteria.

OTHER SPECIAL HANDLING

The following additional considerations apply to the execution of interactive programs.

- An EOR or EOF write on a terminal file has no special significance except that it ensures that the buffer is dumped to the terminal.
- When terminal input data is passed to an executing program, the following convention is followed.

  If the input data consists of data followed by a carriage return, the system supplies an EOR level of one (level number is in bits 17 through 14 of FET+0). If there is no data input but only a carriage return, the system supplies an EOF.

  A program can process input from a terminal properly by sensing an EOR level of 1. The system input/output macros and common decks (except the READO macro) are coded to handle this case properly. Most of the system utilities interact with a terminal in this manner.

  Input from a terminal is handled in the same manner as the CIO READSKP request. If the program's buffer is not large enough to accommodate a full line of input, the data is truncated and the excess is lost. No error code is returned.

- A program that is interacting with a terminal should not do a recall on a file assigned to a terminal if it does an EOR or EOF write on the file to clear the buffer (do not specify the r option on WRITER or WRITEF macros). If recall is specified, an extra rollout of the program may be required before the program terminates. If the conventions mentioned earlier concerning the status of output are followed, it is not mandatory to write an EOR or EOF to clear the buffer; however, if it is done without auto recall (to remain compatible for batch use, for example) as the last operation before the ENDRUN macro, little additional system overhead is incurred.

- If the conventions for special handling are followed, a job being rolled out by the system has all completed lines of data in the output buffer sent to the terminal.

- If a program aborts because an error occurs, the contents of the first message buffer of the control point area are sent to the terminal as part of the output. Messages can be placed in this area using the MESSAGE macro. Messages longer than 48 characters are truncated to 48 characters. If you are in the batch subsystem, this message buffer is always sent to your terminal at the end of each job step, except for job steps that are within a procedure file.

- If a buffer argument error is detected on an output buffer when output is being issued automatically, the output is ignored and the FET is not acted on. This error usually indicates that the executing program has destroyed part of its own field length.

- A COMPASS program can determine whether it is interacting with a terminal input or output file by checking the type of equipment to which the file was assigned (byte zero of FET+1). Refer to the common deck COMCSTF, appendix F.

PROGRAM CONTROL OF TERMINAL ACTIVITY

The remainder of this section describes methods of controlling terminal activity when connected to the Interactive Facility (IAF).

You can control terminal activity in the following ways.

- Include control bytes in your program output to control the positioning of the printing element and define alternate input modes.

- Enter various characters or sequences of characters to control program execution.

- Issue a REPRIEVE macro (refer to section 10) to disable the terminal operator's control of the user program during various phases of execution.

- Issue a CSET macro to change the character set of the terminal.

- Issue a PROMPT macro to control the issuing of the input prompt.

- Issue a TSTATUS macro to determine certain terminal characteristics.

- Issue an EFFECT macro to control the formatting of your terminal output.

CONTROL BYTES

A control byte is a 12-bit quantity, right-justified in bit position 0, 12, 24, 36, or 48 of a CM word.

NOTE

You must ensure that data is not mistaken for a control byte. For example, in the 64-character set, the characters :D at the beginning of a line and followed by an end-of-line will cause you to be logged off if they are written to the terminal output file, since the code 0004 is transmitted.

The following paragraphs describe the bytes available to a user program and their functions. Symbols for most of these control bytes are defined in the system OPL common deck COMSTCM, and may be used by COMPASS programs.

0000—End-of-Line

End-of-line generates a carriage return and line feed, positioning the terminal printing element at the beginning of the next line. An end-of-line consists of 12 to 66 bits of zero, right-justified in one or two central memory words.

0001, 0002—End-of-Block

This byte prevents the positioning of the terminal printing element at the beginning of the next line. An end-of-block byte can be used to allow the terminal user to enter input on the same line as the input request is printed. This byte must be followed by an end-of-line. If not followed immediately by a read request, any output following this byte may be lost.

0003—Auto Input

This byte is used by IAF for auto mode input. The preceding characters in the line in which this byte occurs are sent to the terminal and are also retained as the first characters of the input line. The network can retain a maximum of 20 characters. This byte must be followed by an end-of-line. The next terminal operation must be an input request. The terminal prompt (a question mark) is suppressed.

Refer to Volume 3, System Commands, for further information about auto mode input.

NOTE

The 0003 control byte cannot be used in conjunction with transparent output.

0004—Log Off User

This byte disconnects the terminal from IAF and the network. This byte must be the first byte of a line and must be followed by an end-of-line.

0005—Initiate ASCII Input

This byte causes the next line of input to be translated into 6/12 display code. The terminal prompt (a question mark) is suppressed. This byte must be followed by an end-of-line. On the subsequent input, all terminal control characters or sequences (refer to Control of Program Execution in this section) and terminal definition commands are processed as they are for normal input mode unless the SE terminal definition command has been entered (refer to Volume 3, System Commands). End-of-line (normally a carriage return) terminates 0005 input mode.

If input has been typed ahead when this byte is issued, that input is translated in the old mode.

0006—Initiate Transparent Input Mode

This byte changes the input mode from normal or ASCII to transparent mode. The terminal prompt (a question mark) is suppressed. The byte must be byte 0 of the first word of a line and the word must be terminated by an end-of-line. Bytes 1 and 2 are defined as follows:

Byte 1 — Specifies the maximum number of characters (up to 7777_8) to be received before input is terminated. If byte 1 is 0, the number is assumed to be 1.

Byte 2 — Specifies the termination code. When a character is received from the terminal that matches this byte, the input operation is terminated. Depending on the parity specified for the terminal connection, the lower 8 bits of byte 2 are matched against all 8 bits of the input character, or the lower 7 bits of byte 2 are matched against the lower 7 bits of the input character. If bit 11 of this byte is set, no termination character is assumed.†

This conversion mode packs each 8 bits of data as the lower 8 bits of a 12-bit byte and sets the upper bit (bit 11). The exhaustion of the character count or the occurrence of the termination code causes the end-of-line condition to be set. A 0007 byte is forced as the first byte of input so the data is transmitted as transparent output if it is listed.

A user selected input prompt should follow the 0006 word if it is necessary to guarantee that the terminal is in transparent input mode before input is entered. For example, the following two words, when written to an ASCII-code terminal output file, switch the terminal to transparent input mode and issue three BEL characters (ASCII code 007) at which point transparent input mode is in effect.

If input has been typed ahead when this byte is issued, that input is translated in the old mode.

---

†For more information on parity options, refer to the NAM Version 1/CCP Version 3 Terminal Interface Reference Manual.

```
0006    0012    0015    0000    0000
0007    4007    4007    4007    0000
```

Byte 1 of the 0006 word specifies a maximum character count of 10 (12_8) and byte 2 specifies a termination character of carriage return (015_8 ASCII code). Refer to the description of the 0007 control byte for the second word.

NOTE

The use of transparent input with CYBER Record Manager and/or product set utilities (including compilers) may yield unpredictable results. It is advisable to determine how a particular product set member treats transparent input before proceeding. For example, to initiate Transparent Input from a FORTRAN 5 program, refer to appendix F of the FORTRAN Version 5 Reference Manual for additional information.

0007—Initiate Transparent Output

This byte initiates transparent output. If you wish to output data formatted as described for transparent input, a 0007 byte must precede the data. This mode continues until an end-of-line or nontransparent output data byte is detected. Termination by an end-of-line, however, does not cause a carriage return and line feed. The 0007 byte must be byte 0 (bits 59 through 48) of the first word of a line.

A transparent output data byte is in the format $4xxx_8$, where xxx is the 8-bit octal code for the character being printed. Bits 11 through 8 of a transparent output data byte are always set to 1000_2. If this pattern is not detected, the terminal is switched to the original (normal or ASCII) output mode; this feature can be used to output normal or ASCII data immediately following transparent data without explicitly specifying a termination (end-of-line). The characters 5, 6, 7, and 8 (display codes 40_8, 41_8, 42_8, and 43_8) match the transparent mode bit pattern when in the upper half of a byte and are interpreted as transparent data. Therefore, these characters cannot be used to terminate transparent output. Once transparent output mode is terminated, it remains cleared unless resumed by a 0007 control byte.

The following word, when output from an ASCII-code terminal, produces a line feed and prints a question mark. There is no carriage return or line feed after the question mark.

```
0007    4012    4077    0000    0000
```

The 4012 byte produces a line feed (012 is the ASCII code for line feed) and 4077 is the question mark. The transparent output is terminated by the 24 bits of trailing zeros, constituting an end-of-line. Similar output can be obtained with the following:

```
0007    4012    7100    0000    0000
```

Here 71 is the display code for a question mark and 7100 is a nontransparent data byte. Transparent output is terminated at the question mark and the end-of-line produces a carriage return and line feed.

A control byte that changes the terminal input mode (0003, 0005, or 0006) prevents the system from printing a question mark in response to a program request for input. However, in all other cases, a read request on the input buffer causes the system to print a question mark at the terminal (unless the program has issued a PROMPT OFF macro).

NOTE

The use of transparent output with CYBER Record Manager and/or product set utilities (including compilers) may yield unpredictable results. It is advisable to determine how a particular product set member treats transparent output before proceeding. For example, to initiate Transparent Output from a FORTRAN 5 program, refer to appendix F of the FORTRAN Version 5 Reference Manual for additional information.

0010—Reserved

This byte should not be used.

0011—Initiate ASCII Output

This byte is used to initiate 128-character ASCII output (6/12 display code). This byte must be byte 0 (bits 59 through 48) and applies only to the line currently being output.

0012—Reserved

This byte should not be used.

0013—End-of-String

This byte allows a user to terminate a line of output data without repositioning the terminal carriage. This byte must be followed by an end-of-line (which is ignored) and output continues with subsequent data. The 0013 byte should not be used where the effect of an end-of-line is needed (for example, the end of a logical line on a synchronous terminal).

0014—Internal End-of-Block

This byte is the first byte of a word and is followed by an end-of-line. This byte is reserved by NOS and should not be used since it may cause loss of data.

0015—Executive Auto Input

This byte is used for internal auto input mode. It is reserved by NOS and should not be used since it may cause loss of data.

0016—Terminal Redefinition

This byte enables you to alter the characteristics of a terminal that is connected via IAF. The byte begins a string of 12-bit parameter/12-bit value pairs. The upper bit of each 12-bit segment must be set to prevent a zero value from being interpreted as an end-of-line. The 0016 byte string must be terminated by an end-of-line.

Table 12-1 lists parameters and values allowed for each set of terminal classes. For default values for each terminal class, refer to Volume 3, System Commands. Refer to appendix A for the translation of ASCII characters.

Table 12-1. Terminal Redefinition Parameters (Sheet 1 of 4)

Parameter (Mnemonic)	Number (Octal)	Terminal Classes ①	Value Range (Decimal)	Form of Input
Abort block (AB)	51	1 - 8 (9 - 18) ②	0 - 127	Numerical value for character ③
Blocking factor (BF)	31	1 - 8, 10 - 13, 15 (9, 14, 16, 18)	0 - 20	Decimal number
Break as user break 1 (BR)	63	1 - 3, 5 - 8 (4, 9 - 17)	0 - 1	Yes (1), no (0)
Backspace character (BS)	47	1 - 8 (9 - 18)	0 - 127	Numerical value for character ③
Interruption character (B1)	52	1 - 15 (16, 18)	0 - 127	Numerical value for character ③
Termination character (B2)	53	1 - 15 (16, 18)	0 - 127	Numerical value for character ③
Carriage return idle count (CI)	54	1 - 8 (9 - 18)	0 - 99	Decimal number
	56	1 - 8 (9 - 18)	1	CA (1)
Cancel character (CN)	46	1 - 15, 18 (16, 17)	0 - 127	Numerical value for character ③
Cursor positioning (CP)	107	1 - 3, 5 - 8 (4, 9 - 18)	0 - 1	Yes (1), no (0)
Network control character (CT)	50	1 - 18	0 - 127	Numerical value for character ③
Single message transparent input delimiters (DL) ④	70	1 - 8 (9 - 18)	0 - 1	Character specified (1), not specified (0)
	71	1 - 3, 5 - 8 (9 - 18)	0 - 15	Character count (upper byte)
	72	1 - 3, 5 - 8 (9 - 18)	0 - 255	Character count (upper byte)
	73	1 - 8, 10 - 13, 15 (9, 14, 16 - 18)	0 - 255 ⑤	Numerical value for character ③

① Numbers in parentheses in this column indicate terminal classes for which the parameter is ignored.
② Ignored for packet-switching network (PSN) terminals.
③ Any hexadecimal value except 00 - 02, 20, 30 - 39, 3D, 41 - 5A, 61 - 7A, or 7F.
④ If the value of one of the fields for this parameter is changed, the values of all other fields for this parameter must also be specified.
⑤ Not all values are legal for all terminal classes.

Table 12-1. Terminal Redefinition Parameters (Sheet 2 of 4)

Parameter (Mnemonic)	Number (Octal)	Terminal Classes ①	Value Range (Decimal)	Form of Input
	74	1 - 3, 5 - 8 (9 - 18)	0 - 1	Timeout (1), no timeout (0)
	106	1 - 8, 10 - 13, 15	0	Single message (0)
End-of-block character (EB)	100	1 - 3, 5 - 8, 10 - 13, 15	0 - 127 ②	Numerical value for character ⑤
	101	1 - 3, 5 - 8, 10 - 13, 15	1 - 2 ②	EL (1), EB (2)
	102	1 - 3, 5 - 8, 10 - 13, 15 (9, 14, 16, 17)	0 - 3 ②	NO (0), CR (1), LF (2), CL (3)
End-of-line character (EL)	75	1 - 3, 5 - 8, 10 - 13, 15	0 - 127 ②	Numerical value for character ⑤
	76	1 - 3, 5 - 8, 10 - 13, 15	1 - 2	EL (1), EB (2)
	77	1 - 3, 5 - 8, 10 - 13, 15 (9, 14, 16 - 18)	0 - 3 ②	NO (0), CR (1), LF (2), CL (3)
Echoplex mode (EP)	61	1 - 3, 5 - 8 (4, 9 - 18) ③	0 - 1	Yes (1), no (0)
Full ASCII input (FA)	67	1 - 8, 10 - 13, 15	0 - 1	Yes (1), no (0)
Host availability display (HD)	41	1 - 17	0 - 1	Yes (1), no (0)
Input control (IC)	103	1 - 3, 5 - 8 (4, 9 - 18) ③	0 - 1	Yes (1), no (0)
Input device (IN)	64	1 - 8, 10 - 13, 15	0 - 1	Transparent input (1), not transparent (0)
	65	1 - 8 ④	0 - 2 ②	KB (0), PT (1), BK (2)
Line feed idle count (LI)	55	1 - 8 (9 - 18)	0 - 99	Decimal number
	57	1 - 8 (9 - 18)	1	CA (1)

① Numbers in parentheses in this column indicate terminal classes for which the parameter is ignored.
② Not all values are legal for all terminal classes.
③ Ignored for packet-switching network (PSN) terminals.
④ Not allowed for packet-switching network (PSN) terminals.
⑤ Any hexadecimal value except 00 - 02, 20, 30 - 39, 3D, 41 - 5A, 61 - 7A, or 7F.

Table 12-1. Terminal Redefinition Parameters (Sheet 3 of 4)

Parameter (Mnemonic)	Number (Octal)	Terminal Classes ①	Value Range (Decimal)	Form of Input
Lockout unsolicited messages (LK)	40	1 - 15, 18 (16, 17)	0 - 1	Yes (1), no (0)
Output control (OC)	104	1 - 3, 5 - 8, (4, 9 - 18) ②	0 - 1	Yes (1), no (0)
Output device (OP)	66	1 - 8 (9 - 18)	0 - 2 ③	DI (0), PR (1), PT (2)
Parity processing (PA)	62	1 - 3, 5 - 8	0 - 4	Z (0), O (1), E (2), N (3), I (4)
Page waiting (PG)	45	1 - 8, 10 - 13, 15, 18 (9, 14, 16, 17)	0 - 1	Yes (1), no (0)
Page length (PL)	44	1 - 18	0, 8 - 255 ③	Decimal number
Page width (PW)	43	1 - 18	0, 20 - 255	Decimal number
Special editing mode (SE)	60	1 - 8 (9 - 18) ④	0 - 1	Yes (1), no (0)
Terminal class (TC)	42	1 - 18	1 - 18 ③	Decimal number
Multimessage transparent input delimiter (XL) ⑤	70	1 - 8 (9 - 18)	0 - 1	Character specified (1), not specified (0)
	71	1 - 3, 5 - 8 (9 - 18)	0 - 15	Character count (upper byte)
	72	1 - 3, 5 - 8 (9 - 18)	0 - 255	Character count (lower byte)
	73	1 - 8, 10 - 13, 15 (9, 14, 16 - 18)	0 - 255 ③	Numerical value for character ⑥
	74	1 - 3, 5 - 8 (9 - 18)	0 - 1	Timeout (1), no timeout (0)

① Numbers in parentheses in this column indicate terminal classes for which the parameter is ignored.
② Ignored for PSN terminals.
③ Not all values are legal for all terminal classes.
④ Not allowed for PSN terminals.
⑤ If the value for one of the fields for this parameter is changed, the values of all other fields for this parameter must also be specified.
⑥ Any hexadecimal value except 00 - 02, 20, 30 - 39, 3D, 41 - 5A, 61 - 7A, or 7F.

Table 12-1. Terminal Redefinition Parameters (Sheet 4 of 4)

Parameter (Mnemonic)	Number (Octal)	Terminal Classes ①	Value Range (Decimal)	Form of Input
	105	1 - 8 (9 - 18)	0 - 255 ②	Numerical value for character ③
	106	1 - 8, 10 - 13, 15	1	Multimessage (1)
Transmission block size	36	1 - 18 ④	0 - 7	Decimal number (upper byte)
	37	1 - 18	0 - 255	Decimal number (lower byte)
Upline block limit	30	1 - 18	1 - 31 ②	Decimal number
Full duplex	127	1 - 3, 5 - 8 (4, 9 - 18)	0 - 1	Yes (1), no (0)

① Numbers in parentheses in this column indicate terminal classes for which the parameter is ignored.
② Not all values are legal for all terminal classes.
③ Any hexadecimal value except 00 - 02, 20, 30 - 39, 3D, 41 - 5A, 61 - 7A, or 7F.
④ Ignored for PSN terminals.

Up to 18 terminal characteristic values can be changed by a single 0016 byte string. For example, to change the page width to 100 (144_8), control character to %, and turn the page wait option on, issue the following string.

0016 4043 4144 4050 4045 4045 4001 0000 0000 0000

In the preceding example, each byte has the indicated meaning:

0016	Terminal redefinition control byte.
4043	Specifies page width parameter (43_8).
4144	Sets page width to 100 (144_8).
4050	Specifies control character parameter (50_8).
4045	Sets control character to % (45_8).
4045	Specifies page wait parameter (45_8).
4001	Sets page wait on (1 equals yes).
00...00	End-of-line (terminates terminal redefinition control byte).

For a complete description of terminal characteristics, refer to Volume 3, System Commands.

CONTROL OF PROGRAM EXECUTION

By entering various sequences of keys or commands at your terminal, you can control an interactive program during all phases of execution. The following is a list of the interruption and termination sequences and their effect on the executing program (the ⓒⓡ denotes the message terminator key).

Phase	Sequence †	Effect
Waiting for input	CTRL/T ⓒⓡ	The program is terminated and the terminal is placed in command mode. All other entries are passed to the program as data.
Generating output	BREAK seq	Pressing the BREAK key on asynchronous terminals suspends the output operation. For synchronous terminals, output is automatically suspended on page boundaries, at which point terminal control can be attempted.

NOTE

On asynchronous terminals, output can be suspended by pressing any key that transmits a character. The character entered is also sent to IAF if a ⓒⓡ or line feed is entered. The integrity of this character is not guaranteed, and may cause the interruption sequence which follows to be ignored. It is recommended that the BREAK key be used to suspend output.

† The termination (sometimes referred to as user break 2) sequence (that is, CTRL/Tⓒⓡ) and interruption (sometimes referred to as user break 1) sequence (that is, CTRL/Pⓒⓡ) may vary for different terminals, and may be changed by terminal definition commands. Refer to Volume 3, System Commands, for further information. CTRL refers to the control key. Thus, CTRL/T is entered by keeping the control key depressed while pressing the T key.

Phase	Sequence†	Effect
		Once output is suspended, further execution depends upon the subsequent sequence of keys or characters entered. Each of the following sequences has the indicated effect on the output operation.

seq	Effect
CTRL/T(CR)	This has the same effect as the termination sequence in the input phase.
CTRL/P(CR)††	The output operation ceases and the terminal is placed in suspend mode. At this point IAF recognizes only:

- The P key causing a return to program control, discarding any data in the program buffer prior to the interruption sequence.
- A (CR) causing output to be continued and program control to continue normally. Portions of the output may be lost when output is suspended.
- Any other key (with the exception of terminal control characters and terminal definition commands) is interpreted as a request to terminate the job step.

If any other sequence of characters or keys is entered followed by a (CR), the output operation is resumed with the last partial line repeated.

Executing	CTRL/T(CR) or CTRL/P(CR)	These have the same effect as if they were entered during output phase.

†The termination (sometimes referred to as user break 2) sequence (that is, CTRL/T(CR)) and interruption (sometimes referred to as user break 1) sequence (that is, CTRL/P(CR)) may vary for different terminals and may be changed by terminal definition commands. Refer to Volume 3, System Commands, for further information. CTRL refers to the control key. Thus, CTRL/T is entered by keeping the control key depressed while pressing the T key.

††If the BR terminal redefinition parameter has been set to 1 or if the terminal definition command cBR=Y or TRMDEF,BR=Y has been entered, the BREAK key has the same effect as CTRL/P(CR).

REPRIEVE PROCESSING FOR USER BREAKS 1 AND 2

If the user program has selected terminal interrupt reprieve processing, control will be passed to the user's interrupt package when a user break 1 or 2 is encountered. The user program may now process the interrupt as desired.

To ignore either interrupt, a REPRIEVE RESUME call may be used. This will restart the interrupted sequence as if the interrupt had never occurred.

If the user wishes to have the system process the user break 2, a REPRIEVE RESET call will return control to the system, and the interrupt will be processed in the normal manner.

The user break 1 interrupt cannot be returned to the system by a RESET call.

If the user desires the sequence

```
*INTERRUPTED* (CR)
```

that is normally associated with a user break 1, the sequence will have to be emulated in the user's interrupt handler.

CHARACTER SETS A

A character set is composed of graphic and control characters. A code set is a set of codes used to represent each character within a character set.

A graphic character may be displayed at a terminal or printed by a line printer. Examples are the characters A through Z and the digits 0 through 9. A control character initiates, modifies, or stops a control operation. An example is the backspace character that moves the terminal carriage or cursor back one space. Although a control character is not a graphic character, a terminal may produce a graphic representation when it receives a control character.

All references within this manual to the ASCII character set or the ASCII code set refer to the character set and code set defined in the American National Standard Code for Information Interchange (ASCII, ANSI Standard X3.4-1977). References in this manual to the ASCII character set do not necessarily refer to the ASCII code set.

NOS supports the following character sets.

- CDC graphic 64- (or 63-) character set.
- ASCII 128-character set.
- ASCII graphic 64- (or 63-) character set.
- ASCII graphic 95-character set.

Each installation selects either the 64-character set or the 63-character set. The differences between the two are described in Character Set Anomalies in this appendix. Any reference in this appendix to the 64-character set implies either the 63- or 64-character set, unless otherwise stated.

NOS supports the following code sets.

- 6-bit display code.
- 6/12-bit display code.
- 7-bit ASCII code.

Display code is a set of 6-bit codes from 00_8 to 77_8.

The 6/12-bit display code is a combination of 6-bit codes and 12-bit codes. The 6-bit codes are 00_8 through 77_8, excluding 74_8 and 76_8. (Refer to Character Set Anomalies for the interpretation of the 00_8 and 63_8 codes.) The 12-bit codes begin with either 74_8 or 76_8 and are followed by a 6-bit code. Thus, 74_8 and 76_8 are considered escape codes and are never used as 6-bit codes within the 6/12-bit display code set. The 12-bit codes are 7401_8, 7402_8, 7404_8, 7407_8, and 7601_8 through 7677_8. All other 12-bit codes ($74xx_8$ and 7600_8) are undefined.

The 7-bit ASCII code (as defined by ANSI Standard X3.4-1977) is right-justified in a 12-bit byte. Assuming that the bits are numbered from the right starting with 0, bits 0 through 6 contain the ASCII code, bits 7 through 10 contain zeros, and bit 11 distinguishes the 0000_8 code from the end-of-line byte. The 7-bit codes are 0001_8 through 0177_8 and 4000_8.

CHARACTER SET ANOMALIES

NOS interprets two codes differently when the installation selects the 63-character set rather than the 64-character set. In tables A-1, A-2, and A-3, the codes for the colon and percent graphic characters in the 64-character set are unshaded; the codes for the colon and percent graphic characters in the 63-character set are shaded.

If an installation uses the 63-character set, the colon graphic character is always represented by a 63_8 code, and the 00_8 code is undefined. However, if the installation uses the 64-character set, output of 6/12-bit display codes 7404_8 or 6-bit display code 00_8 produces a colon. In ASCII mode for interactive jobs, a colon can be input only as a 7404_8 6/12-bit display code.

When using either the 63- or 64-character set, the use of undefined 6/12-bit display codes in output files produces unpredictable results and should be avoided.

On input, NOS recognizes alternate 029 punch codes of 11-0 for the right bracket (]) and 12-0 for the left bracket ([). The alternate codes support the COBOL sign overpunch convention and are not recommended for other uses. Refer to COBOL 5 Reference Manual.

Also, two 00_8 codes may be confused with an end-of-line byte and should be avoided (refer to appendix F for further explanation).

Translation of 7-bit ASCII to 6-bit display code causes character set folding from the 128-character ASCII set to the 63- or 64-character ASCII subset. The following special character substitutions occur:

7-Bit ASCII			6-Bit Display Code			7-Bit ASCII	
Code	Character		Code	Character		Code	Character
0140	`		74	@		0100	@
0173	{		61	[		0133	[
0174	|	---------->	75	\	---------->	0134	\
0175	}		62	]		0135	]
0176	~		76	^		0136	^

CHARACTER SET TABLES

This appendix contains character set tables for interactive jobs, batch jobs, and jobs involving magnetic tapes. Table A-1 is for interactive jobs, and table A-2 is for batch jobs. Table A-3 is a conversion table used to cross-reference 7-bit ASCII codes and 6/12-bit display codes and to convert ASCII codes from octal to hexadecimal.

Tables A-4, A-5, and A-6 list the magnetic tape codes and their display code equivalents.

The character set tables are designed so that you can find the character represented by a code (such as in a dump) or find the code that represents a character. To find the character represented by a code, look up the code in the column listing the appropriate code set and then find the character on that line in the column listing the appropriate character set. To find the code that represents a character, you first look up the character and then find the code on the same line in the appropriate column.

INTERACTIVE JOBS

Table A-1 shows the character sets and code sets available to you at an ASCII code terminal. When in NORMAL mode (specified by the NORMAL command†), NOS displays the ASCII graphic 64-character set and interprets all input and output as 6-bit display code. When in ASCII mode (specified by the ASCII command), NOS displays the ASCII 128-character set and interprets all input and output as 6/12-bit display code.

To determine the octal or hexadecimal ASCII code for a character, refer to table A-3. (Certain terminal definition commands require specification of an ASCII code.)

On output, the US code is reserved for network use and defined as an end-of-line. Use of this character, except in transparent mode, causes incorrect formatting and possible loss of output characters.

BATCH JOBS

Table A-2 lists the CDC graphic 64-character set, the ASCII graphic 64-character set, and the ASCII graphic 95-character sets. It also lists the code sets and card punch codes (026 and 029) that represent the characters.

The 64-character sets use display code as their code set; the 95-character set uses 7-bit ASCII code. The 95-character set is composed of all the characters in the ASCII 128-character set that can be printed at a line printer (refer to jobs using Line Printers). Only 7-bit ASCII code files can be printed using the ASCII graphic 95-character set. To print a 6/12-bit display code file (usually created by an interactive job in ASCII mode), you must convert the file to 7-bit ASCII code. To do this, you enter the FCOPY command. The 95-character set is represented by 7-bit ASCII codes 0040_8 through 0176_8.

†Normal and ASCII modes can also be selected via the CSET COMPASS macro.

JOBS USING LINE PRINTERS

The batch character set printed depends on the print train used on the line printer to which the file is sent (refer to section 7). The following are the print trains corresponding to each of the batch character sets.

Character Set	Print Train
CDC graphic 64-character set	596-1
ASCII graphic 64-character set	596-5
ASCII graphic 95-character set	596-6

The characters of the default 596-1 print train are listed in the table A-2 column labeled CDC Graphic (64 Character); the 596-5 print train characters are listed in the table A-2 column labeled ASCII Graphic (64 Character); and the 596-6 print train characters are listed in the table A-2 column labeled ASCII Graphic (95 Character).

If a transmission error occurs when printing a line, the system prints the line again. The CDC graphic print train prints a concatenation symbol (↵) in the first printable column of the repeated listing of the line. The ASCII print trains print an underline (_) instead of the concatenation symbol.

If an unprintable character exists in a line (that is, a 7-bit ASCII code outside the range 0040_8 through 0176_8), the number sign (#) appears in the first printable column of a print line, and a space replaces the unprintable character.

To route and correctly print a 6/12-bit display code file on a line printer with the ASCII graphic 95-character set, you must convert the 6/12-bit display code file to a 7-bit ASCII code file with the FCOPY command (refer to section 9). The resulting 7-bit ASCII file can be routed to a line printer (refer to section 7) but cannot be output at an interactive terminal.

Table A-1. Character Sets for Interactive Jobs (Sheet 1 of 2)

ASCII Graphic (64 Character)	ASCII Character (128 Character)	6-Bit Display Code	6/12-Bit Display Code	7-Bit ASCII Code
: colon		00†		
Display code 00 is undefined at sites using the 63-character set.				
A	A	01	01	0101
B	B	02	02	0102
C	C	03	03	0103
D	D	04	04	0104
E	E	05	05	0105
F	F	06	06	0106
G	G	07	07	0107
H	H	10	10	0110
I	I	11	11	0111
J	J	12	12	0112
K	K	13	13	0113
L	L	14	14	0114
M	M	15	15	0115
N	N	16	16	0116
O	O	17	17	0117
P	P	20	20	0120
Q	Q	21	21	0121
R	R	22	22	0122
S	S	23	23	0123
T	T	24	24	0124
U	U	25	25	0125
V	V	26	26	0126
W	W	27	27	0127
X	X	30	30	0130
Y	Y	31	31	0131
Z	Z	32	32	0132
0	0	33	33	0060
1	1	34	34	0061
2	2	35	35	0062
3	3	36	36	0063
4	4	37	37	0064
5	5	40	40	0065
6	6	41	41	0066
7	7	42	42	0067
8	8	43	43	0070
9	9	44	44	0071
+ plus	+ plus	45	45	0053
- dash	- dash	46	46	0055
* asterisk	* asterisk	47	47	0052
/ slant	/ slant	50	50	0057
(opening parenthesis	(opening parenthesis	51	51	0050
) closing parenthesis	) closing parenthesis	52	52	0051
$ dollar sign	$ dollar sign	53	53	0044
= equal	= equal	54	54	0075
space	space	55	55	0040
, comma	, comma	56	56	0054
. period	. period	57	57	0056

ASCII Graphic (64 Character)	ASCII Character (128 Character)	6-Bit Display Code	6/12-Bit Display Code	7-Bit ASCII Code
# number sign	# number sign	60	60	0043
[opening bracket	[opening bracket	61	61	0133
] closing bracket	] closing bracket	62	62	0135
% percent sign	% percent sign	63	63†	0045
: colon	: colon	63	63	0072
" quote	" quote	64	64	0042
_ underline	_ underline	65	65	0137
! exclamation point	! exclamation point	66	66	0041
& ampersand	& ampersand	67	67	0046
' apostrophe	' apostrophe	70	70	0047
? question mark	? question mark	71	71	0077
< less than	< less than	72	72	0074
> greater than	> greater than	73	73	0076
@ commercial at	@ commercial at	74†	7401	0100
\ reverse slant	\ reverse slant	75	75	0134
^ circumflex	^ circumflex	76†	7402	0136
; semicolon	; semicolon	77	77	0073
	: colon		7404	0072
	% percent		7404	0045
	` grave accent	74†	7407	0140
	a		7601	0141
	b		7602	0142
	c		7603	0143
	d		7604	0144
	e		7605	0145
	f		7606	0146
	g		7607	0147
	h		7610	0150
	i		7611	0151
	j		7612	0152
	k		7613	0153
	l		7614	0154
	m		7615	0155
	n		7616	0156
	o		7617	0157
	p		7620	0160
	q		7621	0161
	r		7622	0162
	s		7623	0163
	t		7624	0164
	u		7625	0165
	v		7626	0166
	w		7627	0167
	x		7630	0170
	y		7631	0171
	z		7632	0172
	{ opening brace	61†	7633	0173
	| vertical line	75†	7634	0174
	} closing brace	62†	7635	0175
	~ tilde	76†	7636	0176
	DEL		7637	0177

†The interpretation of this character or code may depend on its context. Refer to Character Set Anomalies elsewhere in this appendix.

Table A-1. Character Sets for Interactive Jobs (Sheet 2 of 2)

ASCII Graphic (64 Character)	ASCII Character (128 Character)	6-Bit Display Code	6/12-Bit Display Code	7-Bit ASCII Code	ASCII Graphic (64 Character)	ASCII Character (128 Character)	6-Bit Display Code	6/12-Bit Display Code	7-Bit ASCII Code
	NUL		7640	4000		DLE		7660	0020
	SOH		7641	0001		DC1		7661	0021
	STX		7642	0002		DC2		7662	0022
	ETX		7643	0003		DC3		7663	0023
	EOT		7644	0004		DC4		7664	0024
	ENQ		7645	0005		NAK		7665	0025
	ACK		7646	0006		SYN		7666	0026
	BEL		7647	0007		ETB		7667	0027
	BS		7650	0010		CAN		7670	0030
	HT		7651	0011		EM		7671	0031
	LF		7652	0012		SUB		7672	0032
	VT		7653	0013		ESC		7673	0033
	FF		7654	0014		FS		7674	0034
	CR		7655	0015		GS		7675	0035
	SO		7656	0016		RS		7676	0036
	SI		7657	0017		US†		7677	0037

†Reserved for network use. Refer to Character Set Tables in this appendix.

Table A-2. Character Sets for Batch Jobs (Sheet 1 of 3)

CDC Graphic (64 Character)	ASCII Graphic (64 Character)	ASCII Graphic (95 Character)	6-Bit Display Code	6/12-Bit Display Code	7-Bit ASCII Code	Punch Code	
						026	029
: colon†	: colon†		00†			8-2	8-2
Display code 00 is undefined at sites using the 63-character set.							
A	A	A	01	01	0101	12-1	12-1
B	B	B	02	02	0102	12-2	12-2
C	C	C	03	03	0103	12-3	12-3
D	D	D	04	04	0104	12-4	12-4
E	E	E	05	05	0105	12-5	12-5
F	F	F	06	06	0106	12-6	12-6
G	G	G	07	07	0107	12-7	12-7
H	H	H	10	10	0110	12-8	12-8
I	I	I	11	11	0111	12-9	12-9
J	J	J	12	12	0112	11-1	11-1
K	K	K	13	13	0113	11-2	11-2
L	L	L	14	14	0114	11-3	11-3
M	M	M	15	15	0115	11-4	11-4
N	N	N	16	16	0116	11-5	11-5
O	O	O	17	17	0117	11-6	11-6
P	P	P	20	20	0120	11-7	11-7
Q	Q	Q	21	21	0121	11-8	11-8
R	R	R	22	22	0122	11-9	11-9
S	S	S	23	23	0123	0-2	0-2
T	T	T	24	24	0124	0-3	0-3
U	U	U	25	25	0125	0-4	0-4
V	V	V	26	26	0126	0-5	0-5
W	W	W	27	27	0127	0-6	0-6
X	X	X	30	30	0130	0-7	0-7
Y	Y	Y	31	31	0131	0-8	0-8
Z	Z	Z	32	32	0132	0-9	0-9
0	0	0	33	33	0060	0	0
1	1	1	34	34	0061	1	1
2	2	2	35	35	0062	2	2
3	3	3	36	36	0063	3	3
4	4	4	37	37	0064	4	4
5	5	5	40	40	0065	5	5
6	6	6	41	41	0066	6	6
7	7	7	42	42	0067	7	7
8	8	8	43	43	0070	8	8
9	9	9	44	44	0071	9	9
+ plus	+ plus	+ plus	45	45	0053	12	12-8-6
- dash	- dash	- dash	46	46	0055	11	11
* asterisk	* asterisk	* asterisk	47	47	0052	11-8-4	11-8-4

†The interpretation of this character or code may depend on its context. Refer to Character Set Anomalies elsewhere in this appendix.

Table A-2. Character Sets for Batch Jobs (Sheet 2 of 3)

CDC Graphic (64 Character)	ASCII Graphic (64 Character)	ASCII Graphic (95 Character)	6-Bit Display Code	6/12-Bit Display Code	7-Bit ASCII Code	Punch Code 026	Punch Code 029
/ slant	/ slant	/ slant	50	50	0057	0-1	0-1
(opening parenthesis	(opening parenthesis	(opening parenthesis	51	51	0050	0-8-4	12-8-5
) closing parenthesis	) closing parenthesis	) closing parenthesis	52	52	0051	12-8-4	11-8-5
$ dollar sign	$ dollar sign	$ dollar sign	53	53	0044	11-8-3	11-8-3
= equal	= equal	= equal	54	54	0075	8-3	8-6
space	space	space	55	55	0040	no punch	no punch
, comma	, comma	, comma	56	56	0054	0-8-3	0-8-3
. period	. period	. period	57	57	0056	12-8-3	12-8-3
≡ equivalence	# number sign	# number sign	60	60	0043	0-8-6	8-3
[opening bracket	[opening bracket	[opening bracket	61	61	0133	8-7	12-8-2†
] closing bracket	] closing bracket	] closing bracket	62	62	0135	0-8-2	11-8-2†
% percent sign†	% percent sign†	% percent sign†	63†	63†	0045	8-6	0-8-4
: colon	: colon	: colon	63	63	0072	8-2	8-2
≠ not equal	" quote	" quote	64	64	0042	8-4	8-7
↱ concatenation	_ underline	_ underline	65	65	0137	0-8-5	0-8-5
∨ logical OR	! exclamation point	! exclamation point	66	66	0041	11-0	12-8-7
∧ logical AND	& ampersand	& ampersand	67	67	0046	0-8-7	12
↑ superscript	' apostrophe	' apostrophe	70	70	0047	11-8-5	8-5
↓ subscript	? question mark	? question mark	71	71	0077	11-8-6	0-8-7
< less than	< less than	< less than	72	72	0074	12-0	12-8-4
> greater than	> greater than	> greater than	73	73	0076	11-8-7	0-8-6
≤ less or equal	@ commercial at		74			8-5	8-4
≥ greater or equal	\ reverse slant	\ reverse slant	75	75	0134	12-8-5	0-8-2
¬ logical NOT	^ circumflex		76			12-8-6	11-8-7
; semicolon	; semicolon	; semicolon	77	77	0073	12-8-7	11-8-6
		@ commercial at	74†	7401	0100		
		^ circumflex	76†	7402	0136		
		: colon†		7404†	0072		
		% percent sign		7404	0045		
		` grave accent	74†	7407	0140		
		a		7601	0141		
		b		7602	0142		
		c		7603	0143		
		d		7604	0144		
		e		7605	0145		
		f		7606	0146		
		g		7607	0147		

†The interpretation of this character or code may depend on its context. Refer to Character Set Anomalies elsewhere in this appendix.

Table A-2. Character Sets for Batch Jobs (Sheet 3 of 3)

CDC Graphic (64 Character)	ASCII Graphic (64 Character)	ASCII Graphic (95 Character)	6-Bit Display Code	6/12-Bit Display Code	7-Bit ASCII Code	Punch Code	
						026	029
		h		7610	0150		
		i		7611	0151		
		j		7612	0152		
		k		7613	0153		
		l		7614	0154		
		m		7615	0155		
		n		7616	0156		
		o		7617	0157		
		p		7620	0160		
		q		7621	0161		
		r		7622	0162		
		s		7623	0163		
		t		7624	0164		
		u		7625	0165		
		v		7626	0166		
		w		7627	0167		
		x		7630	0170		
		y		7631	0171		
		z		7632	0172		
		{ opening brace	61†	7633	0173		
		| vertical line	75†	7634	0174		
		} closing brace	62†	7635	0175		
		~ tilde	76†	7636	0176		

† The interpretation of this character or code may depend on its context. Refer to Character Set Anomalies elsewhere in this appendix.

Table A-3. ASCII to 6/12-Bit Display Code Conversion (Sheet 1 of 2)

ASCII Character (128 Character)	7-Bit ASCII Code		6/12-Bit Display Code	ASCII Character (128 Character)	7-Bit ASCII Code		6/12-Bit Display Code
	Octal	Hexadecimal			Octal	Hexadecimal	
NUL	4000	00	7640	0	0060	30	33
SOH	0001	01	7641	1	0061	31	34
STX	0002	02	7642	2	0062	32	35
ETX	0003	03	7643	3	0063	33	36
EOT	0004	04	7644	4	0064	34	37
ENQ	0005	05	7645	5	0065	35	40
ACK	0006	06	7646	6	0066	36	41
BEL	0007	07	7647	7	0067	37	42
BS	0010	08	7650	8	0070	38	43
HT	0011	09	7651	9	0071	39	44
LF	0012	0A	7652	: colon††	0072	3A	7404††
VT	0013	0B	7653	: colon	0072	3A	63
FF	0014	0C	7654	; semicolon	0073	3B	77
CR	0015	0D	7655	< less than	0074	3C	72
SO	0016	0E	7656	= equal	0075	3D	54
SI	0017	0F	7657	> greater than	0076	3E	73
				? question mark	0077	3F	71
DLE	0020	10	7660				
DC1	0021	11	7661	@ commercial at	0100	40	7401
DC2	0022	12	7662	A	0101	41	01
DC3	0023	13	7663	B	0102	42	02
DC4	0024	14	7664	C	0103	43	03
NAK	0025	15	7665	D	0104	44	04
SYN	0026	16	7666	E	0105	45	05
ETB	0027	17	7667	F	0106	46	06
				G	0107	47	07
CAN	0030	18	7670				
EM	0031	19	7671	H	0110	48	10
SUB	0032	1A	7672	I	0111	49	11
ESC	0033	1B	7673	J	0112	4A	12
FS	0034	1C	7674	K	0113	4B	13
GS	0035	1D	7675	L	0114	4C	14
RS	0036	1E	7676	M	0115	4D	15
US†	0037	1F	7677†	N	0116	4E	16
				O	0117	4F	17
space	0040	20	55				
! exclamation point	0041	21	66	P	0120	50	20
" quote	0042	22	64	Q	0121	51	21
# number sign	0043	23	60	R	0122	52	22
$ dollar sign	0044	24	53	S	0123	53	23
% percent sign††	0045	25	63††	T	0124	54	24
% percent sign	0045	25	7404	U	0125	55	25
& ampersand	0046	26	67	V	0126	56	26
' apostrophe	0047	27	70	W	0127	57	27
(opening parenthesis	0050	28	51	X	0130	58	30
) closing parenthesis	0051	29	52	Y	0131	59	31
* asterisk	0052	2A	47	Z	0132	5A	32
+ plus	0053	2B	45	[opening bracket	0133	5B	61
, comma	0054	2C	56	\ reverse slant	0134	5C	75
- dash	0055	2D	46	] closing bracket	0135	5D	62
. period	0056	2E	57	^ circumflex	0136	5E	7402
/ slant	0057	2F	50	_ underline	0137	5F	65

†Reserved for network use. Refer to Character Set Tables in this appendix.
††The interpretation of this character or code may depend on its context. Refer to Character Set Anomalies in this appendix.

Table A-3. ASCII to 6/12-Bit Display Code Conversion (Sheet 2 of 2)

ASCII Character (128 Character)	7-Bit ASCII Code		6/12-Bit Display Code	ASCII Character (128 Character)	7-Bit ASCII Code		6/12-Bit Display Code
	Octal	Hexadecimal			Octal	Hexadecimal	
` grave accent	0140	60	7407	p	0160	70	7620
a	0141	61	7601	q	0161	71	7621
b	0142	62	7602	r	0162	72	7622
c	0143	63	7603	s	0163	73	7623
d	0144	64	7604	t	0164	74	7624
e	0145	65	7605	u	0165	75	7625
f	0146	66	7606	v	0166	76	7626
g	0147	67	7607	w	0167	77	7627
h	0150	68	7610	x	0170	78	7630
i	0151	69	7611	y	0171	79	7631
j	0152	6A	7612	z	0172	7A	7632
k	0153	6B	7613	{ opening brace	0173	7B	7633
l	0154	6C	7614	| vertical line	0174	7C	7634
m	0155	6D	7615	} closing brace	0175	7D	7635
n	0156	6E	7616	~ tilde	0176	7E	7636
o	0157	6F	7617	DEL	0177	7F	7637

JOBS USING MAGNETIC TAPE

Coded data to be copied from mass storage to magnetic tape is assumed to be represented in display code. NOS converts the data to external BCD code when writing a coded seven-track tape and to ASCII or EBCDIC code (as specified on the tape assignment command) when writing a coded nine-track tape.

Because only 63 characters can be represented in seven-track even parity, one of the 64 display codes is lost in conversion to and from external BCD code. The following shows the differences in conversion depending on the character set (63 or 64) which the system uses. The ASCII character for the specified character code is shown in parentheses. The output arrow shows how the 6-bit display code changes when it is written on tape in external BCD. The input arrow shows how the external BCD code changes when the tape is read and converted to 6-bit display code.

63-Character Set

6-Bit Display Code		External BCD		6-Bit Display Code
00		16 (%)		00
33 (0)	Output	12 (0)	Input	33 (0)
63 (:)		12 (0)		33 (0)

64-Character Set

6-Bit Display Code		External BCD		6-Bit Display Code
00 (:)		12 (0)		33 (0)
33 (0)	Output	12 (0)	Input	33 (0)
63 (%)		16 (%)		63 (%)

If a lowercase ASCII or EBCDIC code is read from a nine-track coded tape, it is converted to its uppercase 6-bit display code equivalent. To read and write lowercase ASCII or EBCDIC characters, you must use the FCOPY utility (refer to Volume 3, System Commands).

Tables A-4 and A-5 show the character set conversion for nine-track tapes. Table A-4 lists the conversions to and from the 7-bit ASCII character code and 6-bit display code. Table A-5 lists the conversions between the EBCDIC character code and the 6-bit display code. Table A-6 shows the character set conversions between external BCD and 6-bit display code for seven-track tapes.

Table A-4. Nine-Track ASCII Coded Tape Conversion

7-Bit ASCII				6 Bit Display Code		7-bit ASCII				6-Bit Display Code	
Code (Hex)	Char†	Code (Hex)	Char††	Char	Code (Octal)	Code (Hex)	Char†	Code (Hex)	Char††	Char	Code (Octal)
20	space	00	NUL	space	55	3E	>	1E	RS	>	73
21	!	7D	}	!	66	3F	?	1F	US	?	71
22	"	02	STX	"	64	40	@	60	`	@	74
23	#	03	ETX	#	60	41	A	61	a	A	01
24	$	04	EOT	$	53	42	B	62	b	B	02
25	%	05	ENQ	%	63	43	C	63	c	C	03
25	%	05	ENQ	space†††	55	44	D	64	d	D	04
26	&	06	ACK	&	67	45	E	65	e	E	05
27	'	07	BEL	'	70	46	F	66	f	F	06
28	(	08	BS	(	51	47	G	67	g	G	07
29	)	09	HT	)	52	48	H	68	h	H	10
2A	*	0A	LF	*	47	49	I	69	i	I	11
2B	+	0B	VT	+	45	4A	J	6A	j	J	12
2C	,	0C	FF	,	56	4B	K	6B	k	K	13
2D	-	0D	CR	-	46	4C	L	6C	l	L	14
2E	.	0E	SO	.	57	4D	M	6D	m	M	15
2F	/	0F	SI	/	50	4E	N	6E	n	N	16
30	0	10	DLE	0	33	4F	O	6F	o	O	17
31	1	11	DC1	1	34	50	P	70	p	P	20
32	2	12	DC2	2	35	51	Q	71	q	Q	21
33	3	13	DC3	3	36	52	R	72	r	R	22
34	4	14	DC4	4	37	53	S	73	s	S	23
35	5	15	NAK	5	40	54	T	74	t	T	24
36	6	16	SYN	6	41	55	U	75	u	U	25
37	7	17	ETB	7	42	56	V	76	v	V	26
38	8	18	CAN	8	43	57	W	77	w	W	27
39	9	19	EM	9	44	58	X	78	x	X	30
3A	:	1A	SUB	:	00	59	Y	79	y	Y	31
6-Bit display code 00 is undefined at sites using the 63-character set.						5A	Z	7A	z	Z	32
						5B	[	1C	FS	[	61
3A	:	1A	SUB	:	63	5C	\	7C	|	\	75
3B	;	1B	ESC	;	77	5D	]	01	SOH	]	62
3C	<	7B	{	<	72	5E	^	7E	~	^	76
3D	=	1D	GS	=	54	5F	_	7F	DEL	_	65

†When these characters are copied from/or to a tape, the characters remain the same but the codes change from one code set to the other.

††These characters do not exist in 6-bit display code. Therefore, when the characters are copied from a tape, each 7-bit ASCII character is changed to an alternate 6-bit display code character. The corresponding codes are also changed. Example: When the system copies a lowercase a, 61_{16}, from tape, it writes an uppercase A, 01_{8}.

†††A 6-bit display code space always translates to a 7-bit ASCII space.

Table A-5. Nine-Track EBCDIC Coded Tape Conversion

EBCDIC				6-Bit Display Code		EBCDIC				6-Bit Display Code	
Code (Hex)	Char†	Code (Hex)	Char††	Char††	Code (Octal)	Code (Hex)	Char†	Code (Hex)	Char††	Char	Code (Octal)
40	space	00	NUL	space	55	C4	D	84	d	D	04
4A	¢	1C	IFS	[	61	C5	E	85	e	E	05
4B	.	0E	SO	.	57	C6	F	86	f	F	06
4C	<	C0	{	<	72	C7	G	87	g	G	07
4D	(	16	BS	(	51	C8	H	88	h	H	10
4E	+	0B	VT	+	45	C9	I	89	i	I	11
4F	|	D0	}	!	66	D1	J	91	j	J	12
50	&	2E	ACK	&	67	D2	K	92	k	K	13
5A	!	01	SOH	]	62	D3	L	93	l	L	14
5B	$	37	EOT	$	53	D4	M	94	m	M	15
5C	*	25	LF	*	47	D5	N	95	n	N	16
5D	)	05	HT	)	52	D6	O	96	o	O	17
5E	;	27	ESC	;	77	D7	P	97	p	P	20
5F	¬	A1	~	^	76	D8	Q	98	q	Q	21
60	-	0D	CR	-	46	D9	R	99	r	R	22
61	/	0F	SI	/	50	E0	\	6A	|	\	75
6B	,	0C	FF	,	56	E2	S	A2	s	S	23
6C	%	2D	ENQ	%	63	E3	T	A3	t	T	24
6C	¦	2D	ENQ	space†††	55	E4	U	A4	u	U	25
6D		07	DEL		65	E5	V	A5	v	V	26
6E	>	1E	IRS	>	73	E6	W	A6	w	W	27
6F	?	1F	IUS	?	71	E7	X	A7	x	X	30
7A	:	3F	SUB	:	00	E8	Y	A8	y	Y	31
6-Bit display code 00 is undefined at sites using the 63-character set.						E9	Z	A9	z	Z	32
7A	;	3F	SUB	:	63	F0	0	10	DLE	0	33
7B	#	03	ETX	#	60	F1	1	11	DC1	1	34
7C	@	79	\	@	74	F2	2	12	DC2	2	35
7D	'	2F	BEL	'	70	F3	3	13	TM	3	36
7E	=	1D	IGS	=	54	F4	4	3C	DC4	4	37
7F	"	02	STX	"	64	F5	5	3D	NAK	5	40
C1	A	81	a	A	01	F6	6	32	SYN	6	41
C2	B	82	b	B	02	F7	7	26	ETB	7	42
C3	C	83	c	C	03	F8	8	18	CAN	8	43
						F9	9	19	EM	9	44

†When these characters are copied from/or to a tape, the characters remain the same (except EBCDIC codes 4A, 4F, 5A, and 5F) but the codes change from one code set to the other.

††These characters do not exist in 6-bit display code. Therefore, when the characters are copied from a tape, each EBCDIC character is changed to an alternate 6-bit display code character. The corresponding codes are also changed. Example: When the system copies a lowercase a, 81_{16}, from tape, it writes an uppercase A, 01_8.

†††All EBCDIC codes not listed translate to 6-bit display code 55_8 (space). A 6-bit display code space always translates to an EBCDIC space.

Table A-6. Seven-Track Coded Tape Conversions

External BCD	ASCII Character	6-Bit Display Code (Octal)	External BCD	ASCII Character	6-Bit Display Code (Octal)
01	1	34	40	-	46
02	2	35	41	J	12
03	3	36	42	K	13
04	4	37	43	L	14
05	5	40	44	M	15
06	6	41	45	N	16
07	7	42	46	O	17
10	8	43	47	P	20
11	9	44	50	Q	21
12†	0	33	51	R	22
13	=	54	52	!	66
14	"	64	53	$	53
15	@	74	54	*	47
16†	%	63	55	'	70
17	[	61	56	?	71
20	space	55	57	>	73
21	/	50	60	+	45
22	S	23	61	A	01
23	T	24	62	B	02
24	U	25	63	C	03
25	V	26	64	D	04
26	W	27	65	E	05
27	X	30	66	F	06
30	Y	31	67	G	07
31	Z	32	70	H	10
32	]	62	71	I	11
33	,	56	72	<	72
34	(	51	73	.	57
35		65	74	)	52
36	#	60	75	\	75
37	&	67	76	^	76
			77	;	77

†As explained previously in this section, conversion of these codes depends on whether the tape is being read or written.

MESSAGES B

This appendix contains an alphabetical listing of the messages that may appear in your dayfile. Lowercase characters are used to identify a variable names or fields. All messages are sorted according to the first nonvariable word or character. Messages beginning with special characters (such as hyphens or asterisks) are sorted as if the special character(s) were not present. For example, the message

```
filenam ALREADY PERMANENT, AT addr.
```

is listed with the messages beginning with A, and the message

```
**** ENDING SUPPORT OF LFM FUNCTION nnB.
```

is listed with the messages beginning with E.

This appendix contains messages of interest to the applications COMPASS programmer. Messages pertaining to general job processing are contained in appendix B of Volume 3, System Commands.

If you encounter a diagnostic or informative message that does not appear in this appendix, consult the NOS 2 Diagnostic Index. This publication catalogs all messages produced by NOS and its products and specifies the manual or manuals in which each message is fully documented.

MESSAGE	SIGNIFICANCE	ACTION	ROUTINE
ACCESS CATEGORIES NOT VALID FOR JOB.	You attempted to use the SETPFAC command or macro to set one or more permanent file access categories for which you are not validated.	Retry using a set of access categories for which you are validated and that is valid for your job.	PFM
ACCESS LEVEL NOT VALID FOR FILE.	One of the following. - You attempted to save a local file using an access level for the permanent file that is not valid for the device on which the local file resides. - You attempted to define an existing local file as a direct access permanent file using an access level for the permanent file that is not valid for the device on which the local file resides.	Retry using an access level for the permanent file that is not lower than the access level of the local file.	PFM
ACCESS LEVEL NOT VALID FOR FILE filename AT address.	The access level of your job is not valid for the device on which file filename resides.	None. You may not write on this file while your job is at the current access level.	1MS
ACCESS LEVEL NOT VALID FOR JOB.	One of the following: - You attempted to save a local file at an access level that is not valid for your job. - You attempted to define an existing local file as a direct access permanent file at an access level that is not valid for your job. - You attempted to set an access level for a permanent file that is not valid for your job using the SETPFAL command or macro.	Retry using a valid access level.	PFM
ACCESS LEVEL NOT VALID FOR JOB, filename AT address.	The access level specified in the FET for file filename is not within your job's access level limits.	None. You may not create the file at this access level.	1MS
ACCESS LEVEL NOT VALID ON PF DEVICE.	On a SAVE, DEFINE, or SETPFAL command or macro, you specified an access level that is outside the range of access levels allowed on your permanent file device.	Inform security administrator that your permanent file device is not valid for all the access levels for which you are validated.	PFM
ACCOUNT BLOCK LIMIT.	The monitor detected the expiration of the account block SRU limit.	Reset account block SRU limit with SETASL command or macro. If the account block limit is set at its maximum, issue another CHARGE command to begin a new account block.	1AJ
ADDRESS OUT OF RANGE.	An address in a parameter block is outside the job's field length.	Specify parameter block address within field length.	LFM
filename ALREADY PERMANENT.	You have already saved or defined a file with the name specified.	Save or define file using different file name or purge existing file.	PFM
ARG. ERROR.	LDR parameters were outside your job's field length or the central memory address in call is not within your job's field length.	Examine program to determine error.	LDR
ARITHMETIC INDEFINITE.	The CPU floating-point arithmetic unit attempted to use an indefinite operand.	Analyze the job output and dumps to determine the cause.	1AJ
ARITHMETIC OVERFLOW.	The CPU floating-point arithmetic unit received an operand too large for computation.	Analyze the job output and dumps to determine the cause.	1AJ
ARITHMETIC UNDERFLOW.	The CPU floating-point arithmetic unit received an operand too small for computation.	Analyze the job output and dumps to determine the cause.	1AJ
est, ASSIGNED TO filename.	Informative message specifying EST ordinal assigned to file filename.	None.	LFM
ttest, ASSIGNED TO filename.	Equipment type tt with EST ordinal est was assigned to file filename.	None.	LFM

MESSAGE	SIGNIFICANCE	ACTION	ROUTINE
jsn ASSIGNS EXCEED DEMANDS.	RESEX internal problems occurred. The resources actually assigned to the job with the specified job sequence name exceed the resources demanded on a RESOURC command.	Inform site analyst.	RESEX
BAD CATALOG/PERMIT SECTOR.	PFM has encountered a catalog or permit sector which does not have a valid sector length. This indicates corruption of system areas on disk.	Inform site analyst. If many of these errors occur, the site analyst should perform a full PFDUMP, total INITIALIZE, and full PFLOAD on the device.	PFM
BLANK TAPE, filename AT address.	A blank tape was read. Blank tape is defined as more than 25 feet of erased tape.	Ensure correct tape is specified on command.	1MT
BLOCK SEQUENCE ERROR, filename AT address.	The block length recorded in the file did not match the length of the block read, or the block number recorded in the file did not match the system block count (this message applies to I format tapes only).	Ensure accuracy of format parameter (F) on command or macro.	1MT
BLOCK TOO LARGE, filename AT address.	The tape being read contained a data block greater in size than that allowed by the specified format or by your specification.	Ensure accuracy of format parameter (F) on command or macro.	1MT
BOT/EOT ENCOUNTERED, filename AT address.	Indicates an abnormal tape position.	Inform site analyst if persistent.	1MT
BREAKPOINT CONDITION.	The job executed an address for which a breakpoint was requested by the system.	Inform software support.	1AJ
BUFFER ARG. ERROR.	CM address in call is not less than the field length minus the word count; buffer extends past the job's field length.	Verify that operation does not reference address beyond end of buffer or job's field length.	1AJ TCS
BUFFER ARGUMENT ERROR, filename AT address.	For tape operations, this message indicates one of the following. - FET less than 7 words long for S/L format - MLRS greater than 1000 octal for S format - POSMF issued and no HDR1 label found in FET or extended label buffer	Examine program to determine error.	1MT
BUFFER CONTROL WORD ERROR, filename AT address.	Either an attempt was made to write a block smaller than the noise size on an S, L, or F format tape, or a control word error occurred in a write (such as bad byte count).	Examine program to determine error.	1MT
BUFFER CONTROL WORD ERROR ON filename AT address.	Block length specified on WRITECW operation exceeds the allowable physical record size for the device.	Correct the program that is formatting the data for the WRITECW operation.	1MS
pfn BUSY.	The specified direct access file is currently attached in an incompatible mode, or is already being accessed by 4095 users in your specified mode (R, RA, RU, or RM).	Reissue ATTACH until file becomes available, or issue ATTACH specifying WB option.	PFM
CHANNEL MALFUNCTION, filename AT address.	A hardware malfunction occurred.	Inform site analyst.	1MT
CHECKPOINT nnnn COMPLETE.	Indicates that checkpoint nnnn has completed. Issued if only one checkpoint file is present.	None.	CHKPT
CHECKPOINT nnnn COMPLETED TO filename.	Indicates that checkpoint nnnn has been completed to file filename. Issued if alternate CB checkpoint files are used.	None.	CHKPT
CHECKPOINT FILE ACCESS LEVEL ERROR.	In a secure system, the checkpoint file access level is not the same as the job access level limit. This is to insure that the checkpoint file is secure enought to contain all local files. The access level of the checkpoint file is not changed to this level since this is not possible for magnetic tape files or direct access permanent file, and these are the usual file types used for checkpoint files.	Insure that the access level of the checkpoint file is the same as the job access level upper limit.	CHKPT
CHKPT UNABLE TO READ TAPE.	Indicates a tape file had an unrecovered parity error when CHKPT tried to write it to the checkpoint file.	None.	CHKPT
CIO ERROR.	Updating of resource file returned error status other than end-of-device.	Inform site analyst.	RESEX
CKP REQUEST.	A checkpoint has been initiated.	None.	CHKPT

MESSAGE	SIGNIFICANCE	ACTION	ROUTINE
CM BLOCK OUT OF RANGE.	Data transfer from extended memory specified a CM address outside the job field length.	Analyze the job output and dumps to determine the cause.	1AJ
CM OR EC REQUEST EXCEEDS MAXIMUM.	Field length of more than 77000 octal (CM) or more than 7777 octal (EM) was specified in RFL command.	Retry with lower value(s).	CONTROL
CM OUT OF RANGE.	The program referenced an address outside the job CM field length.	Analyze job output and dumps to determine the cause.	1AJ
CM PARITY ERROR.	Double data parity error (two data bits failed) between central memory control (CMC) and CM as detected by the single-error correction double-error detection (SECDED) network, or a single parity error when operating in default mode (SECDED network disabled).	Inform customer engineer.	1AJ
CM RANGE EXIT MODE NOT DESELECTABLE.	You attempted to deselect system checking for CM out of range errors. This cannot be done on a model 176.	Determine if the program can run with CM range error checking. If it cannot, the program must be changed.	CPM
CMC PARITY ERROR.	The CPU sent the central memory control (CMC) data or an address having incorrect parity.	Inform customer engineer.	1AJ
CONNECT REJECT, filename AT address.	The system was unable to connect a peripheral device.	Inform site analyst.	1MT
CONTENTS NOT CHECKPOINTED FOR FILE xxxxxxx	A CHECKPT macro call or a CKP RA+1 call was made and file xxxxxxx is a tape file appearing in the parameter list of the request. The manner in which the file was selected for processing was other than f=4 (position information only). Since magnetic tapes are not copied to the checkpoint file, only the position information was copied (f=4 was forced). Thus, it will be necessary to assign the tape to the restart job. If the contents of the tape is destroyed, it may not be possible to restart the checkpoint job correctly.	Save the tape for the restart job without altering the tape contents from the time of the checkpoint. To eliminate the message, select f=4 for tape files.	RESTART
CONTROL *QAC* ERROR nnn.	A system error was encountered while making a QAC monitor request.	Write a PSR and include support materials to allow CDC to duplicate the problem.	CONTROL
CONTROLLED BACKSPACE ERROR, filename AT address.	Controlled backspace operation failed during write error recovery. Position of tape is uncertain.	Inform customer engineer.	1MT
CPM - ACCESS LEVEL NOT VALID FOR JOB.	You attempted to change the job access level to a value not valid for the job.	Retry with a valid level.	CPM
CPM - ARGUMENT ERROR.	An incorrect argument was specified.	Retry the command using valid arguments.	CPM
CPM - INCORRECT PACK NAME.	An incorrect pack name has been specified.	Ensure that a valid pack name is used.	CPM
CPM - INCORRECT REQUEST.	A job was incorrect CPM function code, subfunction code was specified, or extended reprieve is set (for EREXIT).	Specify auto recall on monitor request call to CPM.	CPM
CPM - INVALID PAGE VALUE.	There was an error in your job page size information.	Retry with correct values.	CPM
CPM - LIBRARY NOT FOUND = library.	The system could not find the specified library.	Ensure that the library name is correct on the command or macro.	CPM
CPM - MISSING *SHELL* LOAD OPTION.	You attempted to set a shell control without specifying a load option.	Specify a load option in the shell command or macro and try again.	CPM
CPM - SECONDARY USER COMMANDS DISABLED.	You entered a secondary USER command on a system that has had secondary USER commands disabled.	None.	CPM
CPM - SYSTEM ERROR.	For function 42 (Special Charge) or 106 (Set Job Characteristics), CPM was unable to read the input file.	Write a PSR and include support materials to allow CDC to duplicate the problem.	CPM

MESSAGE	SIGNIFICANCE	ACTION	ROUTINE
CPM - USER ACCESS NOT VALID.	1. You tried to perform a CPM operation for which you are not authorized. 2. A CPU program issued a DSDOUT or a DSDINP macro request when it did not have access to the L display CMR buffers.	Ensure that the L display utility was initiated at the system console with the appropriate commands.	CPM
CPU ERROR EXIT AT address.	The errors listed after this message occurred at address, causing job termination.	Refer to the descriptions of the error messages issued with this message.	1AJ
CVL CALL ERROR.	If CVL was called by a user via GETMC, CVL encountered one of the following errors while processing the GETMC macro. - Recall bit was not set. - Illegal function number was specified. - Insufficient or improper combination of parameters was specified. If CVL was called by a CE via CEVAL, CVL encountered one of the following errors while processing the CEVAL macro. - Recall bit was not set. - Illegal function number was specified. - Insufficient or improper combination of parameters was specified. - The CE supplied mnemonic does not match the preassigned tape mnemonic when CVL was called by preassignment. - A tape was not preassigned when CVL was called by preassignment. - The CE attempted to load tape controlware when CVL was called by preassignment. - A labeled tape was preassigned when CVL was called by preassignment.	Correct error and rerun program.	CVL
DATA/PERMIT ERRORS.	When the specified file was loaded from tape, errors were encountered in both data and permit information.	Enter CHANGE command or macro with CE parameter to allow access to the file. Make the file local and check if data is accurate. Enter a CATLIST command to see if the permits are accurate.	PFM
DATA TRANSFER ERROR.	An error occurred in a read operation during a file transfer.	Inform site analyst. If many of these errors occur, the site analyst should perform a full PFDUMP, total INITIALIZE, and full PFLOAD on the device.	PFM
DBest,DNdn,message.	A form of PFM error message (issued to the system dayfile, error log, and sometimes the user dayfile) identifying the mass storage equipment on which the error occurred. The equipment type DB is 885-42 disk (full track). est EST ordinal of device. dn Device number. message PFM error message.	Refer to the significance and action of the message as given in this list of error messages.	PFM
DEFAULT FAMILY USED.	You entered a FAMILY command with no parameter. The default family was used.	None.	CONTROL
DEMAND EXCEEDED.	You attempted to assign more units than were scheduled on the RESOURC command.	Increase appropriate parameter value on RESOURC command.	RESEX
DEMAND FILE ERROR.	Resource execution error was encountered. This error occurred because the demand file (RSXDid) entry does not match the job identification.	Inform site analyst.	RESEX
DEVICE ERROR ON FILE filename AT address.	An irrecoverable error occurred on the mass storage device containing the file filename.	Inform customer engineer.	1MS

MESSAGE	SIGNIFICANCE	ACTION	ROUTINE
DEVICE UNAVAILABLE.	One of the following: - A packname was specified for a pack that is not currently mounted. - For a DIS job, no SUI or USER command has been entered. - No permanent file device could be found for your user name. - On a secure system, no permanent file device with the proper access level could be found for your user name.	If a packname was specified, try again with the WB or NA parameter to request that the pack be mounted. For other errors, inform site operator.	PFM
DIest,DNdn,message.	A form of PFM error message (issued to the system dayfile, error log, and sometimes the user dayfile) identifying the mass storage equipment on which the error occurred. The equipment type DI is 844-21 disk (half track). est EST ordinal of device. dn Device number. message PFM error message.	Refer to the significance and action of the message as given in this list of error messages.	PFM
DIRECT ACCESS DEVICE ERROR.	One of the following occurred. - An incorrect device type was specified. - The device on which the local file resides may not contain direct access files.	Specify correct device type, or return the local file and retry.	PFM
DIRECT ACCESS FILE ERROR.	The system sector for the file contains incorrect data or cannot be read.	Inform site anlyst. If many of these errors occur, the site analyst should perform a full PFDUMP, total INITIALIZE, and full PFLOAD on the device.	PFM
DJest,DNdn,message.	A form of PFM error message (issued to the system dayfile, error log, and sometimes the user dayfile) identifying the mass storage equipment on which the error occurred. The equipment type DJ is 844-41/44 disk. est EST ordinal of device. dn Device number. message PFM error message.	Refer to the significance and action of the message as given in this list of error messages.	PFM
DKest,DNdn,message.	A form of PFM error message (issued to the system dayfile, error log, and sometimes the user dayfile) identifying the mass storage equipment on which the error occurred. The equipment type DK is 844-21 disk (full track). est EST ordinal of device. dn Device number. message PFM error message.	Refer to the significance and action of the message as given in this list of error messages.	PFM
DLest,DNdn,message.	A form of PFM error message (issued to the system dayfile, error log, and sometimes the user dayfile) identifying the mass storage equipment on which the error occurred. The equipment type DL is 844-41/44 disk (full track). est EST ordinal of device. dn Device number. message PFM error message.	Refer to the significance and action of the message as given in this list of error messages.	PFM
DMest,DNdn,message.	A form of PFM error message (issued to the system dayfile, error log, and sometimes the user dayfile) identifying the mass storage equipment on which the error occurred. The equipment type DM is 885-11/12 disk (half track). est EST ordinal of device. dn Device number. message PFM error message.	Refer to the significance and action of the message as given in this list of error messages.	PFM
DQest,DNdn,message.	A form of PFM error message (issued to the system dayfile, error log, and sometimes the user dayfile) identifying the mass storage equipment on which the error occurred. The equipment type DQ is 885-11/12 disk (full track). est EST ordinal of device. dn Device number. message PFM error message.	Refer to the significance and action of the message as given in this list of error messages.	PFM
DSP - CANNOT ROUTE JOB INPUT FILE.	An attempt was made to route the job input file.	Copy job input file to a local file and route it.	DSP
DSP - COMPLETE BIT ALREADY SET.	The complete bit was not cleared before DSP was called.	Clear complete bit before calling DSP.	DSP

MESSAGE	SIGNIFICANCE	ACTION	ROUTINE
DSP - DEFERRED ROUTING NOT ALLOWED.	A file cannot be deferred routed if data declaration, or implicit/explicit remote text is specified.	Correct and retry.	DSP
DSP - DEVICE FULL.	There is no space on the device for current use.	Retry route at a later time.	DSP
DSP - DEVICE UNAVAILABLE.	DSP attempted to create a file on a device that was turned off or is currently unavailable for access.	Specify different device or contact site operator.	DSP
DSP - FILE NAME ERROR.	An attempt was made to create a file with an incorrect file name.	Specify valid file name.	DSP
DSP - FILE NOT ON MASS STORAGE.	An attempt was made to route a file not on mass storage.	Copy file to mass storage before routing.	DSP
DSP - FILE ON REMOVABLE DEVICE.	A file on a removable device cannot be routed.	Copy file to nonremovable device before routing.	DSP
DSP - FOMSM CODE NOT ALPHANUMERIC.	The forms code specification (FC=fc parameter) must consist of two alphanumeric characters.	Specify alphanumeric forms code.	DSP
DSP - I/O SEQUENCE ERROR.	A request was made on a busy file.	Wait until file is not busy.	DSP
DSP - IMMEDIATE ROUTING - NO FILE.	The specified file for the immediate routing could not be found.	Ensure that file to be routed is available to job for processing.	DSP
DSP - INCORRECT DISPOSITION CODE.	Specified disposition code is not recognized.	Verify disposition code.	DSP
DSP - INCORRECT EXTERNAL CHARACTERISTICS.	You specified an undefined external characteristic code.	Verify external characteristic code.	DSP
DSP - INCORRECT FILE TYPE.	The file being processed is not a queued file or a local file.	Ensure the file being processed is of the correct type.	DSP
DSP - INCORRECT INTERNAL CHARACTERISTICS.	You specified an undefined internal characteristics code.	Verify internal characteristics code.	DSP
DSP - INCORRECT JOB ABORT CODE.	Value of abort code field of DSP parameter word four is incorrect.	Verify abort code.	DSP
DSP - INCORRECT ORIGIN TYPE.	The system cannot queue the file for input with the specified origin type.	Specify valid origin type.	DSP
DSP - INCORRECT REQUEST.	One of the following: - Parameter block is not within field length. - Request was not made with auto recall.	Change parameter block addresses or specify auto recall.	DSP
DSP - INCORRECT SERVICE CLASS	The two-character service class is not valid for the user or the origin type of the file.	Correct and retry.	DSP
DSP - INCORRECT SPACING CODE.	Value specified for spacing code was greater than 77 octal.	Verify spacing code.	DSP
DSP - INCORRECT TID.	One of the following. - Username and familyname parameters were not in CM field length. - Terminal identifier (TID) is greater than or equal to IDLM for batch jobs. - The familyname or username is not valid.	Verify that TID parameters are valid.	DSP
DSP - INVALID FILE MODE.	The file being processed is attached in execute-only mode.	Return the file and reattach it in other than execute-only mode.	DSP
DSP - INVALID USER CARD.	For an operation on an input file, either a USER command was not included, or the user name or password on USER command is not valid.	Verify that a USER command is present and that the user name and password are correct.	DSP
DSP - MASS STORAGE ERROR.	A mass storage error was encountered on the file.	Write a PSR and include support materials to allow CDC to duplicate the problem.	DSP
DSP - QUEUED FILE READ ERROR.	DSP encountered an error while trying to read the system sector for a previously deferred routed file.	Attempt to recreate the output file.	DSP
DSP - ROUTE TO INPUT NOT IMMEDIATE.	You attempted to queue a file for input using a deferred ROUTE.	Change to immediate route.	DSP

MESSAGE	SIGNIFICANCE	ACTION	ROUTINE
DSP - THIS ROUTING NOT ALLOWED.	You attempted to change the queue disposition of a deferred routed file.	Rescind prior routing by using the SC disposition code with the ROUTE command or macro. Route the file again with the desired final disposition code.	DSP
DSP - TOO MANY DEFERRED BATCH JOBS.	You attempted to submit more deferred batch jobs than allowed by your validation limit.	Wait for jobs to complete or request a larger validation limit from site personnel.	DSP
DSP - UNDEFINED SERVICE CLASS.	The two-character service class is not defined.	Correct and retry.	DSP
DSP - USER ACCESS NOT VALID.	Caller is not validated to specify one of the following: DLID SLID Data declaration	Write a PSR and include support materials to allow CDC to duplicate the problem.	DSP
DUPLICATE FILE NAME.	The file specified already exists in the system.	Use different name in request.	LFM
DUPLICATED LINES.	Duplicate lines, which would have been printed during a dump operation, were suppressed.	None.	CPMEM
DVest,DNdn,message.	A form of PFM error message (issued to the system dayfile, error log, and sometimes the user dayfile) identifying the mass storage equipment on which the error occurred. The equipment type is 819 disk (single density). est EST ordinal of device. dn Device number. message PFM error message.	Refer to the significance and action of the message as given in this list of error messages.	PFM
DWest,DNdn,message.	A form of PFM error message (issued to the system dayfile, error log, and sometimes the user dayfile) identifying the mass storage equipment on which the error occurred. The equipment type DW is 819 disk (double density). est EST ordinal of device. dn Device number. message PFM error message.	Refer to the significance and action of the message as given in this list of error messages.	PFM
EC NOT VALIDATED.	The number of extended memory blocks specified on the Job command either exceeds that for which you are authorized or exceeds the machine size available to user jobs.	Check your validation with the LIMITS command.	CPM
ECS LOAD ERROR.	An error occurred during ECS load.	Inform site analyst.	1AJ
filename EMPTY.	The file specified on a SAVE command contains no data.	Verify that file contains data and retry.	PFM
END OF TAPE, filename AT address.	The end-of-tape was encountered.	Ensure that correct file manipulation operation is specified.	1MT
ENTRY POINT NOT FOUND.	The specified entry point could not be found.	Verify that entry point is valid.	1AJ
EOI CHANGED BY RECOVERY.	The file was truncated during deadstart recovery.	Use CHANGE command or macro with CE parameter to allow access to the file. Make the file local and list it to determine how much of the file was lost.	PFM
EQUIPMENT NOT AVAILABLE.	The requested equipment is either in use or does not exist, or no requested equipment with the proper access level can be found.	Ensure accuracy of macro or command or retry at a later time.	LFM RESEX
ERASE LIMIT, filename AT address.	The system made 20 erasures (10 feet of tape) without being able to successfully write the tape.	Clean tape or use different tape.	1MT
ERROR CODE ec, filename AT address.	1MT error code ec has occurred but no specific message is issued. This would normally not occur unless the job was dropped by the operator.	Inform site analyst.	1MT

MESSAGE	SIGNIFICANCE	ACTION	ROUTINE
ERROR - FILES(S) NOT PROCESSED.	One or more files were not checkpointed because CHKPT detected an error while getting a list of the files at the control point. This could occur if there are many local files.	Contact Central Software Support.	CHKPT
ERROR FLAG TERMINATION, filename AT address.	The job was aborted with a tape operation in progress. The operation/request is not complete. For example, the operator could kill the job while tape error recovery is in progress.	None.	1MT
ERROR FLAG TERMINATION, FILE filename AT address	The job was aborted with an input/output request in progress. The operation/request is not complete.	None.	1MS
ERROR IN FILE DATA.	When the specified file was loaded from tape, an error was encountered in the data.	Enter CHANGE command or macro with CE parameter to allow access to the file. Make the file local and ensure the accuracy of the data.	PFM
ERROR IN PERMIT DATA.	When the specified file was loaded from tape, an error was encountered in the permit entries.	Enter CHANGE command or macro with CE parameter to allow access to the file. Do a CATLIST to see if the permits are accurate.	PFM
ERROR IN ROUTE FUNCTION, LFN=filename.	Informative message issued to the system dayfile indicating that an error occurred while routing filename.	Examine the job's dayfile for a more specific error message.	DSP
ERROR ON FILE - PROFILa.	Either the profile file cannot be found or there is a bad profile file level-3 block random address.	Inform site analyst.	CPM
EXCHANGE PACKAGE/MEMORY DUMP ON FILE ZZZDUMP.	The exchange package and memory dump is written on local file ZZZDUMP because the job is of interactive origin and file OUTPUT is assigned to a terminal.	To examine the exchange package and dump, list file ZZZDUMP.	CPMEM
EXTENDED MEMORY BLOCK OUT OF RANGE.	Data transfer between CM and extended memory specified an extended memory address outside the job field length.	Analyze the job output and dumps to determine the cause of the error.	1AJ
EXTENDED MEMORY FLAG REGISTER PARITY.	Parity error detected on extended memory flag register operation.	Inform customer engineer.	1AJ
EXTENDED MEMORY OUT OF RANGE.	Job referenced extended memory address outside job field length.	Analyze the job output and dumps to determine the cause of the error.	1AJ
EXTENDED MEMORY PARITY ERROR.	ECS transfer on a model 176 failed.	Inform customer engineer.	1AJ
FAMILY NOT FOUND.	No family exists with the name you specified.	Retry with an existing family name.	CONTROL
FET ADDRESS OUT OF RANGE AT address.	FET extends past job's field length or FET address .LE. 1.	Verify that FET address is valid.	1MS
FET PARAMETER ERROR ON filename AT address.	One of the parameters in the FET is incorrect or the FET is not long enough for the parameter.	Verify that FET is valid.	1MS
FET TOO SHORT.	FET is not long enough for processing by requested function.	Increase FET size.	LFM
FIELD BEGINNING AT nn NO COMPARE.	Verification of label field at character position nn does not compare with that specified by the user on an OPEN statement.	None.	1MT
FILE BOI/EOI/UI MISMATCH.	Data in the system sector for file filename does not match information from the EOI sector and/or catalog information. This indicates that the file has been destroyed.	Inform site analyst. If many of these errors occur, the site analyst should perform a full PFDUMP, total INITALIZE, and full PFLOAD on the device.	PFM
FILE EMPTY.	The file specified was empty.	Verify that the file contains data and retry.	LFM SFM
FILE ERROR filename.	An incorrect address was detected on file filename.	Correct error and retry.	RESTART
FILE/JOB NOT FOUND.	The specified file or job was not found in the system.	Use the ENQUIRE command to ensure job is still in the system.	CONTROL

MESSAGE	SIGNIFICANCE	ACTION	ROUTINE
FILE LENGTH ERROR.	The length of a file does not equal the length specified in the catalog. This indicates that the file has been destroyed.	Inform site analyst. If many of these errors occur, the site analyst should perform a full PFDUMP, total INITIALIZE, and full PFLOAD on the device.	PFM
FILE NAME ERROR.	File name contains incorrect characters or contains more than seven characters.	Ensure that a valid file name is specified.	LFM LISTLB PFM
FILE NOT DISK RESIDENT.	Access to a file which does not reside on disk is not permitted with the requesting function.	Write a PSR.	PFM
FILE NOT ON MASS STORAGE.	The specified file does not reside on mass storage.	Copy file to mass storage and retry.	1AJ
FILE TOO LONG.	The local file specified for a SAVE, REPLACE, or APPEND command exceeds the length allowed or the direct access file specified for an ATTACH operation in WRITE, MODIFY, UPDATE, or APPEND mode exceeds the direct access file length limit for which you are authorized.	Reduce length of file or save as a direct access file.	PFM
FILE TOO LONG, FILE filename AT address.	The length of the direct access permanent file currently being written exceeds the direct access file length limit for which you are authorized.	Reduce the length of the file or ask site personnel for a larger direct access file length limit.	1MS
FL REQUEST BEYOND MFL (ECS).	Extended memory field length requirements for the job step exceed the extended memory field length allowed.	Increase job step extended memory field length.	1MA
FL TOO SHORT FOR PROGRAM.	The job's field length is too short for the program.	Rerun the job with larger field length specification.	1AJ 1AJ
FLE TOO SHORT FOR LOAD.	You attempted to load extended memory data beyond the job's extended memory field length.	None.	1AJ
FNT FULL.	FNT space is currently not available for a request that has both the ep and up processing options set.	Return some files and retry operation.	PFM
FORMAT REQUIRES UNLABELED TAPE.	The format specified (F) is valid only for unlabeled tapes.	The tape must be assigned as an unlabeled tape.	RESEX
FUNCTION REJECT, filename AT address.	Function was rejected (possible hardware problem).	Inform site analyst.	1MT
FWA .GE. LWA+1.	The first word address parameter was greater than the last word address parameter on DMP, DMD, DMPECS, DMDECS, LOC, or PBC command or DMP, DMD, DED, or DEP system requests.	Correct error and retry.	CPMEM
FWA/LWA .GE. FL.	Either first word address parameter of LOC or the last word address of LOC or PBC was greater than or equal to your field length.	Reduce FWA and/or LWA and retry.	CPMEM
GETFNT TABLE TOO LARGE.	The table generated by the GETFNT macro is too large to fit in the job's field length.	Specify larger GETFNT table size.	LFM
HTIME NOT AVAILABLE.	The HTIME command or macro is valid only on a model 176.	None.	1AJ
I/O ON EXECUTE-ONLY FILE filename AT address.	Your job issued a CIO function not allowed on an execute-only file.	Attach file in proper mode to allow the desired operation.	1MS
I/O SEQUENCE ERROR.	Action was requested on a local file that was busy.	Write a PSR.	LFM PFM 1AJ
I/O SEQUENCE ERROR ON FILE filename AT address.	You attempted to perform a function on a local file before another function on the same file was complete.	Modify program to wait until each function is complete before attempting another.	1MS
ILLEGAL ACCESS LEVEL FOR EQUIPMENT.	Your have specified a level outside of the equipment access level limits.	Use access level within required equipment's access level limits, or use equipment with access level required.	RESEX

MESSAGE	SIGNIFICANCE	ACTION	ROUTINE
IMPROPER ACCESSIBILITY.	You did not specify the correct file accessibility on the LABEL command or macro, or volume accessibility was set and you attempted to assign the tape as unlabeled from a nonsystem origin job.	Ensure accuracy of request.	RESEX
INCORRECT APPLICATION ACCOUNTING REQUEST.	The application program that issued this message attempted to initiate application accounting incorrectly.	Inform data base administrator.	CPM
INCORRECT COMMON MEMORY MANAGER REQUEST.	A memory request was issued with reserved bits in the parameter block set.	Do not set reserved bits.	1MA
INCORRECT DEVICE REQUEST.	The device type (R parameter) specified on a request for an auxiliary device cannot be recognized or does not exist in the system. If the auxiliary device specified by the PN parameter is not the same type as the system or job default, the R parameter must be included.	Examine auxiliary device request and ensure its accuracy.	PFM
INCORRECT END OF JOB OPTION SPECIFIED.	The option specified is not valid for a noninteractive job.	Retry with a different option.	CPM
INCORRECT EQUIPMENT.	Equipment specified does not exist or is not allowed (for example, an interactive teminal is requested from other than interactive origin, or a tape is being requested with the REQUEST macro).	Ensure that file resides on a legal equipment type.	LFM RESEX
INCORRECT EQUIPMENT ON FILE filename AT address.	A system error.	Inform site analyst.	1MS
INCORRECT ERROR EXIT ADDRESS.	Error exit address is beyond the job's current field length.	None.	1AJ
INCORRECT EXTERNAL CALL.	RESEX did not recognize an external call.	Inform site analyst.	RESEX
INCORRECT FILE MODE.	You tried to unlock a file that was attached in execute-only mode.	None.	LFM
INCORRECT FILE NAME filename AT address.	The file name does not conform to established rules. The file name must have a maximum of seven characters, alphanumeric, and must be left-justified with binary zero fill.	Use valid file name.	1MS
INCORRECT FILE STATUS.	Specified file status is incorrect.	Check value.	LFM
INCORRECT FILE TYPE.	The specified file is of a type not allowed in the requested operation. Possible causes include attempts to - Change a nonlocal file to file type library. - Designate a direct access file as the primary file. - Route the primary file. - Change the access level of a nonlocal file.	Verify that file type is appropriate.	LFM
pfn INCORRECT FILE TYPE.	The operation requested cannot be performed on the specified file because it is of the wrong file type. This message is issued when you attempt to define a direct access file with the same local file name as a file currently assigned to the job that is of a file type other than local.	Define the file using a unique name or return the conflicting file.	PFM
INCORRECT I/O REQUEST ON FILE filename AT address.	The system could not recognize the specified function code, or the code was not valid for the type of device to which the file was assigned. You attempted to write on a local file that resides on a device whose highest access level is below the current access level of your job. The system provides a dump of the FET on file OUTPUT.	Verify CIO function code being used. Lower the access level of your job, if possible, or raise the access level of the file.	1MS
INCORRECT ID CODE.	An identification code not in the range 0-67B or 77B is present on the LDI command.	Reissue the request with the correct identification code.	LDI
INCORRECT INSTRUCTION.	The CPU attempted to execute an incorrect or unavailable instruction.	Analyze job output and dumps to determine the cause of the error.	1AJ
INCORRECT LABEL TYPE, filename AT address.	Incorrect label type. The only valid label types are ANSI labeled and nonstandard labeled.	Use correct label type.	1MT

MESSAGE	SIGNIFICANCE	ACTION	ROUTINE
INCORRECT LOAD ADDRESS.	The load address is less than 2.	Specify larger load address and retry.	1AJ
INCORRECT MODIFICATION OF filename AT address.	Either you have attempted to shorten a modify-only file or the file cannot be modified at all.	Determine whether file can be modified.	1MS
INCORRECT RANDOM ADDRESS.	The current random address specified is zero.	Verify current random address.	LFM
INCORRECT USER ACCESS.	You are not authorized to access a restricted subsystem without proper validation.	Determine proper validation requirements.	EXINIT
INCORRECT USER COMMAND.	The user name or password could not be validated, or a secondary USER command to a different family was attempted, and you are not validated to change families.	Verify that user name and password are valid.	CPM
INDEX ADDRESS OUT OF RANGE FOR filename AT address.	The random sector address for a random input/output request was equal to or greater than the job's field length.	Correct random sector address.	1MS
INDEX TOO LARGE ON OPEN, FILE filename AT address.	For an OPEN function, the random index length exceeded the index buffer size.	Set EP bit in FET+1 to get error status without abort or increase index buffer size.	1MS
INPUT FILE IN NO RERUN STATUS.	Informative message.	None.	QFM
INPUT FILE IN RERUN STATUS.	Informative message.	None.	QFM
INSUFFICIENT RESOURCES ON SYSTEM.	Resource demand exceeds the number of units physically available on the system.	Reduce resource demand.	RESEX
INTERLOCK NOT AVAILABLE.	A software interlock is currently not available to the requesting function with both ep and up processing options specified.	Retry function at a later time.	PFM
pfn INTERLOCKED.	The file is currently not available for WRITE, MODIFY, UPDATE, or APPEND modes of access because a system utility has attached the file in utility mode.	Retry operation later.	PFM
INVALID FILE ACCESS LEVEL.	The access level specified in the FET for an LFM request is not valid for that user, or for the equipment on which the file resided.	Correct the access level and retry.	LFM
pfn IS DIRECT ACCESS.	An indirect access file operation was attempted on a direct access file.	Use the appropriate direct access file request.	PFM
filename IS EXECUTE ONLY.	The specified local file cannot be accessed by SAVE, REPLACE, APPEND or DEFINE since it is an execute-only file.	None.	PFM
pfn IS INDIRECT ACCESS.	A direct access file operation was attempted on an indirect access file.	Use the appropriate indirect access file request.	PFM
ITF, CONNECTING TO HOST pid.	Indicates state of connection to host pid.	None.	ITF
JOB CANNOT ACCESS FILE.	The permanent file being processed has either an access level or access categories that are not valid for your job.	None. Your job can not access the file.	PFM
JOB CARD ERROR (jobcmmnd)	The Job command of a routed job contains an error. jobcmmnd is the first 20 characters of the Job command in error.	Correct the Job command and retry.	DSP
JOB HUNG IN AUTORECALL.	The address to which the job was recalled does not have the complete bit set, and there is no system activity outstanding that could set the complete bit.	Analyze job and dumps to determine cause.	1AJ
JOB IN NORERUN STATE ON RECOVERY.	Identifies a job recovered on level 0 or level 3 deadstart that was aborted because it was in a no-rerun mode (due to NORERUN command or macro).	Refer to the NORERUN command or macro description.	1AJ
JOB NOT RERUN.	The system was unable to successfully rerun a job because of a mass storage read error or because the QFT is full.	Resubmit the job to the system.	1AJ
JOB REPRIEVED.	The job has been successfully reprieved.	None.	1AJ
JOB RERUN.	The job has been terminated and requeued for input.	None.	1AJ

MESSAGE	SIGNIFICANCE	ACTION	ROUTINE
JOB STEP EXCEEDS ACCOUNT BLOCK.	You tried to set your job step limit to a value greater than your account block limit or tried to set your account block limit to a value less than your job step limit.	Check values on SETJSL and SETASL commands.	CPM
JOB STEP LIMIT.	The monitor detected the expiration of the job step SRU limit.	Reset job step limit with SETJSL command or macro and retry. If job step SRU limit is set at maximum, request increased SRU validation.	1AJ
LABEL CONTENT ERROR, filename AT address.	A block read was the correct size for a label but one or more required fields (such as the label name) were incorrect.	Use LISTLB command to obtain label data and retry.	1MT
LABEL MISSING, filename AT address.	During a read operation, a required label was missing.	Ensure that tape has label.	1MT
LABEL NOT EXPIRED.	You attempted to write on a tape with an unexpired label.	If current contents of tape can be sacrificed, have operator blank label tape. Otherwise, wait until label has expired.	1MT
LABEL PARAMETER ERROR ON OPEN, filename AT address.	Label fields did not match on open request. An additional message, FIELD BEGINNING AT addr NO COMPARE. specifying the decimal character position in HDR1 of the first field that did not compare correctly, is also issued.	Use LISTLB command to obtain label data.	1MT
LDR ERROR.	Issued before one of the following errors. - OVERLAY NOT FOUND IN LIBRARY. - ARGUMENT ERROR.	Correct error and retry.	LDR
LFM DETECTED ERROR.	The Resource Executive (RESEX) detected an error.	Inform site analyst.	LFM
LFM ERROR.	LFM detected an error in the request. See other error messages in dayfile for more information.	Ensure that a valid LFM request is being made.	LFM
LFM INCORRECT REQUEST.	One of the following: - The function detected was not recognized as a valid LFM function. - An LFM function was issued without the auto recall bit set.	Verify that a valid LFM request is being used.	LFM
LOADER MISSING.	1AJ could not find BEGIN or LDR=.	Inform software support.	1AJ
LOCAL FILE LIMIT.	The job tried to create too many local files.	Reduce local file usage or see site administrator about changing your validation limits.	1AJ
M.T. NOT AVAILABLE ON FILE filename AT address.	The magnetic tape executive is not executing.	Inform site operator. Bring up MAGNET.	1MS
MAGNETIC TAPE SUBSYSTEM NOT ACTIVE.	One of the following occurred during a magnetic tape request. - There was no UDT address in the FST. - The UDT address in the FST was incorrect. - The MAG subsystem was not present.	Contact site operator to have MAGNET brought up. Otherwise contact central software support.	LFM RESEX
MASS STORAGE ERROR.	An error was encountered in reading a portion of the permanent file catalog or permit information (error log and dayfile message). This indicates a hardware problem with a disk pack or disk drive.	Inform site analyst. If many of these errors occur, the site anlayst should perform a full PFDUMP, total INITIALIZE, and full PFLOAD on the device.	PFM
MESSAGE LIMIT.	The number of messages issued by the job has exceeded the limit for which you are authorized. Message functions issued by compilers or applications that run in the job's field length are also counted as job dayfile messages and thus are subject to your dayfile message limit.	Split job into two or more jobs and retry.	1AJ
MFL LESS THAN EXTENDED MEMORY MINIMUM CM FL.	To use extended memory you must have a required minimum amount of central memory field length. This message indicates that you do not have the required central memory field length.	Increase central memory field length.	1MA
MISSING DEMAND FILE ENTRY.	Dayfile message indicating a RESEX internal problem. The overcommitment algorithm was initiated without a demand file entry having been defined previously.	Inform site analyst.	RESEX

MESSAGE	SIGNIFICANCE	ACTION	ROUTINE
jsn MISSING RESOURCE.	Dayfile message indicating that a RESEX internal problem occurred. RESEX expected but did not find a resource unit assigned to the job with job sequence number jsn. This could occur if the MAG subsystem was stopped while tapes were assigned.	Inform site analyst.	RESEX
MISSING VSN OR EQUIPMENT ASSIGNMENT.	Dayfile message indicating internal malfunction in RESEX (expected VSN or equipment assignment was not found).	Inform site analyst.	RESEX
MLF REQUEST TOO SMALL, MINIMUM USED.	The SETMFL macro (CPM function 52) attempted to specify a maximum FL value lower than the minimum required for a subsequent MFL command.	No action necessary. To prevent the message from recurring increase MFL specified in SETMFL macro to at least the minimum required for the MFL command.	CPM
MONITOR CALL ERROR, xxxx nnnnnnnnn.	One of the following occurred during the processing of an monitor request: - The monitor request call was not recognized by the system. - Bad parameters were supplied in the request. - Incorrect user access was encountered. - An incorrect request was included in the request.	Examine program to determine why incorrect monitor request call is being made.	1AJ
MT,Ccc,Eec,Hhhhhhhhh,B.C. RESTART.	Magnetic tape controller controlware restarted.	None.	1MT
MT,Ccc,Eec,Hhhhhhhhh,BAD ERASE.	Error detected after an erase was attempted to recover a write error.	Inform site analyst.	1MT
MT,Ccc,Eec,Hhhhhhhhh,BID RECOVERY-x.	A single block mispositioning error was recovered by block ID recovery. If x is B, the error was caused by backspacing the tape too far; if x is F, the tape was not backspaced far enough.	None.	1MT
MT,Ccc,Eec,Hhhhhhhhh,BLOCK TOO LARGE.	Data block is at least one byte longer than length bbbb shown in third line of message.	None.	1MT
MT,Ccc,Eec,Hhhhhhhhh,BUSY.	Unit was still busy after one second.	Inform customer engineer.	1MT
MT,Ccc,Eec,Hhhhhhhhh,CHANNEL ILL.	Channel is not accepting function for status requests properly.	Inform customer engineer.	1MT
MT,Ccc,Eec,Hhhhhhhhh,CON. REJ.	Connect reject; unable to connect to the unit.	Inform site analyst.	1MT
MT,Ccc,Eec,Hhhhhhhhh,CON. REJ. MDI.	Connect reject; unable to connect to unit because of marginal detection indication (thermal warning). Unit turned off.	Inform customer engineer.	1MT
MT,Ccc,Eec,Hhhhhhhhh,CON. REJ. OFF.	Connect reject; unable to connect to unit. Unit turned off.	Inform site analyst.	1MT
MT,Ccc,Eec,Hhhhhhhhh,FNffff,Pyyyy.	Function ffff was rejected by the controller; yyyy is the address in 1MT where the function was initiated.	Inform site analyst.	1MT
MT,Ccc,Eec,Hhhhhhhhh,Lbbbb,Bnnnnnn.	The length (bbbb) and block number (nnnnnn) read from trailer bytes in block did not match the actual length or the block number read given in previous message line.	None.	1MT
MT,Ccc,Eec,Hhhhhhhhh,LOAD CHECK.	Load sequence failed on the unit.	Push CLEAR button and reload tape, or inform site analyst.	1MT
MT,Ccc,Eec,Hhhhhhhhh,MARGINAL, DOWN.	Indicates controller failure. Channel has been logically turned off and maintenance is required.	Inform customer engineer.	1MT
MT,Ccc,Eec,Hhhhhhhhh,MARGINAL, OFF.	Unit has been logically turned off because of read/write failure. This occurred when a special function to check the read/write path to a unit failed during initial label scan. Maintenance is required.	Inform customer engineer.	1MT
MT,Ccc,Eec,Hhhhhhhhh,NO EOP.	No end-of-operation detected from unit within 1 second.	Inform customer engineer.	1MT
MT,Ccc,Eec,Hhhhhhhhh,NOISE.	A noise block was skipped on the tape.	None.	1MT
MT,Ccc,Eec,Hhhhhhhhh,NOT READY.	Tape unit dropped ready status.	Make unit ready.	1MT
MT,Ccc,Eec,Hhhhhhhhh,ON THE FLY.	Error was corrected as the data was read.	None.	1MT

MESSAGE	SIGNIFICANCE	ACTION	ROUTINE
MT,Ccc,Eec,Hhhhhhhhh,POSITION LOST.	The last good block written cannot be found during write recovery.	None.	1MT
MT,Ccc,Eec,Hhhhhhhhh,RECOVERED.	Previously reported error has been successfully recovered.	None.	1MT
MT,Ccc,Eec,Hhhhhhhhh,STATUS.	Error type cannot be determined so actual controller status is returned.	Inform site analyst.	1MT
MT,Ccc,Eec,Hhhhhhhhh,WRONG PARITY.	Tape was written in parity opposite that being read.	None.	1MT
MT,Ccc-e-uu,vsn,rw,est,Ss,GSgggggggg MT,Ccc,Dddd...d. MT,Ccc,Uuu...u,Ttttt. MT,Ccc,Aaaaaaaaa. MT,Ccc,Fff,Iii,Bnnnnnn,Lbbbb,Pppppppp. MT,Ccc,Eec,Hhhhhhhhh,type. or MT,Ccc-e-uu,vsn,rw,est,Ss,GSgggggggg. MT,Ccc,Dddd...d. MT,Ccc,Aaaaaaaaa. MT,Ccc,Fff,Iii,Bnnnnnn,Lbbbb,Pppppppp. MT,Ccc,Eec,Hhhhhhhhh,type. or MT,Ccc,-e-uu,vsn,rw,est,Ss,GSgggggggg. MT,Ccc,Dddd...d. MT,Ccc,Fff,Iii,Bnnnnnn,Lbbbb,Pppppppp. MT,Ccc,Eec,Hhhhhhhhh,type. or MT,Ccc-e-uu,vsn,rw,est,Ss,GSgggggggg. MT,Ccc,Dddd...d. MT,Ccc,Uuu...u,Ttttt. MT,Ccc,Fff,Iii,Bnnnnnn,Lbbbb,Pppppppp. MT,Ccc,Eed,Hhhhhhhhh,type.	Four or five or six-line message describing a magnetic tape hardware malfunction on a 66x or 67x tape unit. Message as illustrated indicates 7-track, model 667 or 677 unit. If NT appears in place of MT, message indicates 9-track, model 669 or 679 unit. Message is issued to error log and dayfile. The first line of each message provides the following information. cc-e-uu — Channel, equipment (tape controller), and physical unit number of tape unit on which error was encountered. vsn — Volume serial number associated with tape on the specified unit. rw — Read (RD) or write (WR) operation; any operation not involving an actual read or write is listed as a read. est — EST ordinal of the unit on which the tape was written. This is provided only for labeled tapes generated under NOS; otherwise, the field is blank. s — Channel status. gggggggg — General status of magnetic tape unit. Last byte is block ID. The MT,Ccc,Dddd...d line of the message provides the following information. cc — Channel number; the channel number is repeated to allow the analyst to associate this message with the first message if errors are occurring on more than one tape channel at the same time. ddd...d — Detailed status of magnetic tape unit. The MT,Ccc,Uuu...u,Ttttt line of the message provides the following information. cc — Channel number; repeated to associate this message with the previous message. uu...u — Detailed unit status. tttt — Third byte of the tape unit format parameters (refer to the magnetic tape subsystem reference manual for descriptions of unit format parameter fields). The MT,Ccc,Aaaaaaaa line (for FSC only) contains the additional sense byte status not placed in the detailed status or unit status fields. cc — Channel number. aaaaaaa — Sense byte. The MT,Ccc,Fff,...,Pppppppp line of the message provides the following information. cc — Channel number; repeated to associate this message with the previous message. ff — Software function on which the error occurred. ii — Error iteration; number of times error has been encountered on this unit without successful recovery. nnnnnn — Block number on which error occurred. bbbb — Length of block on which error occurred in octal bytes. pppppppp — 1MT internal error parameters.	Refer to the separate listing of the last line message (MT,...,type.) for the appropriate action.	1MT

MESSAGE	SIGNIFICANCE	ACTION	ROUTINE
	The last line of each message provides the following information. cc — Channel number; repeated to associate this message with the previous message ec — Octal error code value. hhhhhhhh — Parameters passed to the tape unit for the format function (refer to the tape drive's hardware reference manual for descriptions of the unit format parameter fields). type — Additional description of the error. Refer to individual listing of the last line message.		
MT/NT CONFLICT.	Conflict exists between 7-track and 9-track tape descriptors (track type, density, and conversion mode). For example, a request for a 9-track tape specified 200 bpi density. This message can also be issued if the device type specified in FET+1 conflicts with the track type specified in bit 56 of FET+10 octal. If dt=MT and bit 56 is set, the message is issued.	Ensure accuracy of command.	RESEX BLANK
MULTI-FILE NOT FOUND, filename AT address.	Either LISTLB has reached the end of the multifile set or the requested file was not found. The following additional messages are also given. - REQUESTED SECTION xxxx. - FOUND SECTION yyyy. or - FILE IDENTIFIER NOT FOUND. The filename, address, xxxx, and yyyy given can be ignored.	If LISTLB reached the end of the multifile set, the operation is complete and no action is required. Otherwise, ensure that the correct tape is being used and that it contains the desired file(s). All label parameters must match in order to position to the specified file.	1MT
NO EXTENDED MEMORY.	A DMPECS or DMDECS command or DED or DEP request was entered and no extended memory field length is assigned to you.	None.	CPMEM
NO INPUT FILE FOUND.	No valid input file exists; functions cannot be performed.	Verify that input file is present.	QFM
NO MASS STORAGE AVAILABLE.	No mass storage space of the requested type could be found.	Retry later.	LFM
filename - NO TEMP DEVICE FOUND.	You attempted to create a local copy of an indirect access permanent file, but the file's access level was not within the range of valid access levels for any active temporary storage device.	Inform security administrator. The security administrator needs to configure the system so that temporary devices are available for all valid access levels.	PFM
NO WRITE ENABLE, filename AT address.	Either you attempted to write on a tape mounted with no write ring, or no write was allowed because of additional constraints described in an additional message line.	If no additional message line appears, ensure the inserting of a write ring by specifying the W processing option on the tape request (for example, PO=W on the LABEL command). Otherwise, refer to the description of the message in the additional message line.	1MT
NON-MATCHING CONVERSION.	Informative message indicating conversion mode on labeled 9-track tape differs from that specified by assignment request. System writes tape in specified mode, or reads tape with write ring out in correct mode. However, reading tape with write ring in or using wrong conversion mode generates conversion errors.	If reading tape with write ring in, return and reassign with correct conversion mode.	RESEX
NON-MATCHING DENSITY.	Informative message indicating that the density specified on the command or macro is not the same as the density of the assigned tape. Issued only to 9-track tapes with write ring out. 9-track tapes are read at the current density on tape. They are written at specified density if write initiated from load point; otherwise, tape is written at the current density on the tape.	None.	RESEX

MESSAGE	SIGNIFICANCE	ACTION	ROUTINE
NORERUN/RERUN INCORRECT FROM INTERACTIVE JOBS.	You entered a NORERUN or RERUN command from an interactive terminal. The command is ignored.	None.	QFM CONTROL
filename NOT FOUND. or username NOT FOUND.	One of the following: - The specified permanent file could not be found. - The specified user name could not be found. - You are not allowed to access the specified file. - The specified local file could not be found.	Verify that file name/ user name is correct, that access permission has been granted, and that correct access is being attempted.	PFM
filename NOT IN WRITE MODE.	The file specified in an OVWRITE command or macro was not processed because you do not have write access to it.	Correct and retry.	1MS
filename NOT ON MASS STORAGE.	The file to be saved is not on mass storage; the first track of the file is not recognizable.	Verify that file is on mass storage.	PFM
filename NOT ON MASS STORAGE.	The file specified in an OVWRITE command or macro was not processed because it does not reside on mass storage.	Correct and retry.	1MS
NOT VALID TO DOWNGRADE DATA.	You attempted to lower the access level of a permanent file, but you are not validated to downgrade files.	None.	PFM
NOT VALIDATED FOR REQUESTED ACCESS LEVEL.	User has specified an access level outside the user's access level validation.	Use a validated access level, or site must validate user for additional access level.	RESEX
NOT VALIDATED FOR WRITING UNLABELED TAPES.	User has not been validated for writing on unlabeled tape.	Either use labeled tape or site must validate user for SAV=CULT MODVAL privilege.	RESEX
NOT VALIDATED TO SET XD/XT.	You specified an XD or XT parameter in a CHANGE, DEFINE or PERMIT command or macro without having validation to use these parameters.	Retry without XD or XT parameter, or obtain validation and retry.	PFM
ONLY CORRECT PARAMETERS ARE *ON* OR *OFF*.	The parameter you specified was not recognized.	Check command description and retry.	CONTROL
OPERATOR DROP.	Informative message indicating that the operator dropped the job.	None.	1AJ
OPERATOR KILL.	The operator entered a KILL command to drop the job. This disallows erexit processing. A job with extended reprieve processing is reprieved once. Exit processing is allowed.	Correct job as needed and rerun.	1AJ
OPERATOR OVERRIDE.	The operator entered an OVERRIDE command to drop the job. This disallows exit, erexit, and reprieve processing.	Correct job as needed and rerun.	1AJ
OVERLAY FILE EMPTY.	No data appears in the requested file.	Verify that overlay file is valid.	1AJ
OVERLAY FILE NOT FOUND.	The specified file was not available.	Verify that file is local to job and retry.	1AJ
OVERLAY LOST.	The specified overlay was not found.	Inform software support.	1AJ
OVERLAY NOT FOUND.	The specified overlay was not found.	Verify that the file with the specified overlay is local to the job.	1AJ
OVERLAY NOT FOUND IN LIBRARY - ovlname.	The overlay ovlname was not found in the system library.	Verify that call is to valid overlay.	LDR
PARAMETER BLOCK BUSY.	The completion bit of the parameter block was not set to one before the monitor request was made.	Set completion bit before making monitor request call.	LFM
PARAMETER *ON* OR *OFF* REQUIRED.	No parameter was specified on ERRMSG command.	Retry with a parameter.	CONTROL
PARAMETERS *ON* AND *OFF* ARE MUTUALLY EXCLUSIVE.	You specified both the ON and OFF parameters on the same command. Only one is allowed.	Retry with either ON or OFF.	CONTROL
PARITY ERROR, JOB HUNG.	A CM parity error occurred in the jobs field length. The job is hung to freeze the affected area.	Inform software support.	1AJ

MESSAGE	SIGNIFICANCE	ACTION	ROUTINE
pfn PERMANENT ERROR.	The specified direct access file resides on alternate storage and has data errors that must be corrected. The error flag must be cleared or the file must be reloaded from a backup copy.	Inform site analyst. The site analyst may be able to correct the problem with ASDEBUG or SSDEBUG; otherwise, the file must be purged or reloaded.	PFM
PERMIT LIMIT EXCEEDED.	A PERMIT command or macro cannot be processed because the file specified already has the maximum number of permits allowed.	None.	PFM
PF STAGING DISABLED.	The specified direct access file cannot be accessed since it resides on the Mass Storage Facility (MSF) and the site has temporarily disabled all MSF file staging.	Determine from site operator when MSF file staging will resume and retry the job at that time.	PFM
PF UTILITY ACTIVE.	The operation was not attempted because a permanent file utility was currently active.	Wait until PF utility is not active and retry.	PFM
PFM ARGUMENT ERROR.	One of the following conditions is true: - The PFM call block is outside of the job's field length. - An incorrect command code was specified. - Auto recall was not set when PFM was called.	Correct your program to call PFM properly.	PFM
PFM EXCESS ACTIVITY.	The PFM activity count for your permanent file family is currently at its limit. This condition is reported to the requesting program only if both ep and up processing options are set.	Retry operation later.	PFM
PFM INCORRECT REQUEST.	One of the following occurred: - A privileged command was requested by a nonvalidated program. - An incorrect backup requirement, preferred residence, or subsystem value was entered. - Incorrect command code, passed to PFM. - Incorrect permit mode or catalog type specified. - The FET for a file specified in a SETPFAC request was less than six words long.	Verify that the PFM request is valid.	PFM
PL ERROR IN DECK.	The program library specified by the P option in the KRONREF command was incorrectly formatted.	Check that the PL is a MODIFY OPL.	KRONREF
POSITION ERROR ON-filename.	File filename was not repositioned after being checkpointed because CHKPT detected an address error.	None.	CHKPT
POSITION LOST, filename AT address.	During write or read error recovery, the system could not find the last good block of data, making it impossible to successfully perform error recovery. Labels are not written after this error and existing data on the tape is not destroyed.	Retry operation on different tape or tape drive, if possible.	1MT
PP CALL ERROR.	The monitor detected an error in a CPU request for PP action.	Verify that correct PP call is issued.	1AJ
PRIOR TAPE ASSIGNMENT LOST.	Magnetic tape executive has been dropped along with tapes assigned. All of the job's prior tape assignments are lost.	From an interactive job, return/unload all prior tapes and reassign. A batch job aborts and must be rerun.	RESEX
PROGRAM STOP.	The system processed a program stop (00) instruction.	None.	1AJ
PROGRAM STOP AT address.	The monitor detected a program stop instruction at the specified address.	None.	1AJ
PRU LIMIT.	The job's mass storage PRU limit was exceeded during preparation of a local copy of an indirect access file.	Return one or more local files and retry.	PFM
PRU LIMIT, FILE filename AT address.	The job's mass storage PRU limit was exceeded during an attempt to write or extend this file.	Return one or more local files and retry.	1MS
PRUS REQUESTED UNAVAILABLE.	On a DEFINE, no device currently has available the amount of space requested by the S parameter. On a secured system, no device with the proper access level currently has the amount of space requested.	If possible, reduce the number of PRUs specified by the S parameter, or retry at another time when space might be available.	PFM

MESSAGE	SIGNIFICANCE	ACTION	ROUTINE
QAC BUFFER ARGUMENT ERROR.	The address in one of the QAC buffer fields (first, in, out, or limit) is incorrect.	Ensure that all four addresses are within the job's field length.	QAC
QAC INCORRECT PARAMETER BLOCK SIZE.	The value of the len field of address+1 is incorrect for the requested function.	Correct the len field or the function code.	QAC
QAC INCORRECT QUEUE ORDINAL.	The ordinal in the ejt/qft ord field of address+4 is out of range for the specified queue.	Enter a valid EJT or QFT ordinal.	QAC
QAC INCORRECT REQUEST.	One of the following occurred: - An undefined function code was entered in address+0. - An incorrect address was entered in monitor request or the link address field in address +10B, on the QAC parameter block. - The range of security access levels specified was not within the job's security access level limits. - The new security access level specified in an ALTER request was not within the security access level limits for the file's origin type, or is not allowed on the device where the file resides. - An ALTER request to change the access level of a queued file was made by a user without security administrator privileges.	Enter a valid function code, level(s), address, or access.	QAC
QAC REQUEST COMPLETION BIT SET.	The completion bit in address+0 was set for the request.	Clear the completion bit.	QAC
QFM ARGUMENT ERROR.	One of the following: - Address is outside field length. - Address is equal to 1. - Origin code is out of range. - ID code is out of range.	Verify that QFM request is valid.	QFM
QFM BUFFER ARGUMENT ERROR.	A buffer pointer did not conform to the following constraints. - FIRST .LE. IN - FIRST .LE. OUT - OUT .LE. LIMIT .LE. FL	Examine program to determine error in buffer pointers.	QFM
QFM FILE NAME ERROR.	The filename specified is not a valid file name.	Verify file name.	QFM
QFM I/O SEQUENCE ERROR.	Action was requested on a busy file.	Wait until file is not busy and retry.	QFM
QFM INCORRECT EQUIPMENT.	The equipment specified in FET+7 either is not mass storage or is not in the range of the EST.	Verify that correct equipment is specified in FET+7.	QFM
QFM INCORRECT QUEUE FILE ORDINAL.	The QFT ordinal of the file to be attached is out of range.	Ensure that the routine calling QFM supplies the correct ordinal.	QFM
QFM INCORRECT REQUEST.	One of the following. - Specified function was incorrect or undefined. - Job did not have SSJ= entry point. - Auto recall bit was not set.	Verify that valid QFM request is being made.	QFM
QFM USER ACCESS NOT VALID.	The user tried to perform an operation for which he is not validated (for example, attempting to run a system origin job from nonsystem origin).	Ensure accuracy of command or macro or determine proper validation requirements.	QFM
RANDOM ADDRESS NOT ON FILE filename AT address.	The random address specified was not within the bounds of the file. The system provides a dump of the FET on file OUTPUT.	Verify that random address is correct.	1MS
RANDOM INDEX ERROR.	The random disk address of the permit sector is in error (error log and dayfile message). This may be a problem with your program.	Check your program. If it does not use the CATLIST macro, inform site analyst.	PFM
READ AFTER WRITE, filename AT address.	You attempted to read a tape on which the last operation was a write.	Ensure accuracy of tape positioning commands (BKSP, BKSPRU, SKIPFB, or REWIND required to read after write).	1MT
READY DROP, filename AT address.	Unit dropped ready status.	None.	1MT

MESSAGE	SIGNIFICANCE	ACTION	ROUTINE
REPLACE ERROR.	The same file was found twice during a catalog search. This error can occur for APPEND or REPLACE commands or macros after a file is found and purged and the catalog search is continued (error log and dayfile message).	Inform site analyst. If many of these errors occur, the site analyst should perform a full PFDUMP, total INITIALIZE, and full PFLOAD on the device.	PFM
REPRIEVE BLOCK ERROR.	An address is out of range or there is an incorrect parameter in the reprieve parameter block at the time of an error. The message is also issued if the specified reprieve address itself is out of range. (1AJ issues this message for all errors except terminal interrupts processed by 1RI.)	Ensure parameter block is correct.	1AJ
REPRIEVE CHECKSUM BAD.	The computed checksum does not agree with the checksum specified in the parameter block at the time of the error. (1AJ issues this message for all errors except terminal interrupts processed by 1RI.)	Ensure interrupt handler is still intact. Ensure that code in the area for which checksum was computed has not changed.	1AJ
REQUEST *K* DISPLAY.	Issued by function 5 (Set Console Display Register) to DSD B and J displays when operator action is requested.	Refer to DSD K display.	CPM
REQUEST UNDEFINED ON DEVICE, filename AT address.	The specified function cannot be performed on the device on which the file resides. The system provides a dump of the FET on file OUTPUT.	Verify that valid device is specified.	1MS
RESEX ABORT - OPERATOR TERMINATION.	The operator entered a DROP, KILL or RERUN command, setting an error flag in RESEX. RESEX performed appropriate cleanup procedures before termination.	Determine reason for operator action. Rerun job if possible.	RESEX
RESEX ABORT - SYSTEM RESOURCE LIMIT.	RESEX terminated prematurely due to job time limit, SRU limit, or track limit. RESEX performs appropriate cleanup procedures before termination.	If error caused by SRU or time limit, increase resource limits. If caused by track limit, contact site analyst.	RESEX
RESEX ABORT - TERMINAL INTERRUPT.	Terminal user interrupted RESEX (interrupt or terminate sequence). RESEX performs appropriate cleanup procedures before termination.	None.	RESEX
RESEX FAILURE.	The resource executive (RESEX) has detected a fatal error.	Write a PSR.	PFM
RESOURCE ENVIRONMENT ERROR.	Dayfile message indicating RESEX internal problem occurred (internal environment building failed due to MST, UDT, or EST errors).	Inform site analyst.	RESEX
RESOURCE NEGATIVE SHARE COUNT.	Dayfile message indicating that a RESEX internal problem occurred. The resource overcommitment algorithm indicates a greater number of users are sharing a removable pack than are actually sharing the pack.	Inform site analyst.	RESEX
RESOURCE PF ERROR ec filename.	PFM error ec occurred when attaching resource file filename.	Inform site analyst.	RESEX
RESOURCE SCRATCH FILE ERROR.	Dayfile message indicating RESEX internal problem has occurred. An empty entry has been found in the overcommitment algorithm scratch file.	Inform site analyst.	RESEX
RFL BEYOND MFL.	The RFL request is greater than the maximum field length for a job step.	Increase maximum field length with MFL command or SETMFL macro.	CPM
ROLLOUT FILE BAD.	A job could not be rolled out correctly.	Inform site analyst. Check error log dayfile for the job that was aborted and the location of the bad rollout file.	1RI
RPV-ADDRESS OUT OF RANGE.	An address provided to RPV was out of range. This includes the following conditions. - FWA of parameter block .GE. FL - LWA of parameter block .GE. FL - Transfer address .GE. FL - Transfer address .LT. 2.	Check all addresses sent to RPV and ensure that they are within range.	RPV
RPV-CALLED WITHOUT AUTO-RECALL.	An extended RPV mode call was made without auto recall.	Ensure that auto recall bit is set.	RPV

MESSAGE	SIGNIFICANCE	ACTION	ROUTINE
RPV-INCORRECT CALL.	One of the following: - An attempt was made to do a reset when no previous error existed or the previous error was pending. - An extended mode call was attempted when nonextended RPV or DISTC had already been set for the job. - A nonextended mode call was attempted when RPV extended mode had already been set in for the job. - A call was made to clear extended reprieve and there were pending interupts present.	- Check to see if a reset is allowed. - If RPV nonextended mode has already been used, then RPV nonextended mode cannot be used in the same job. - Clear pending interrupt f % before calling RPV. - If RPV extended mode has already been used, then RPV nonextended mode cannot be used in the same job.	RPV
RPV-INCORRECT FUNCTION CODE.	The function code specified in the parameter block was incorrect.	Function code in parameter block must equal 1 for setup, 2 for resume, or 3 for reset.	RPV
RPV-INCORRECT PENDING INTERRUPT.	A bit was set in the pending interrupt word of the parameter block which did not correspond to a defined error.	Check parameter block to ensure that all bits set correspond to a defined error.	RPV
RPV-INCORRECT PENDING RA+1 REQUEST.	The pending monitor request field of the parameter block contained an incorrect monitor request on a RESUME or SETUP. On a RESET, the pending monitor request is not validated because the job will not be restarted.	Check pending monitor request word in parameter block to ensure that it contains a valid call.	RPV
RPV-LWA OF CHECKSUM .LT. FWA.	The last word address of the checksum was less than the first word address.	Correct and rerun.	RPV
RPV-PARAMETER BLOCK TOO SHORT.	The length specified in the parameter block was less than 31B words.	Ensure that the length specified in the parameter block is greater than or equal to 31B.	RPV
RPV-UNDEFINED MASK.	The mask specified in the parameter block is not a legal mask.	Check mask set in parameter block to ensure that it is a valid mark.	RPV
SCP INCORRECT TRANSFER ADDRESS.	The SCP SSCR (RA + 51B) word contains incorrect parameters on the completion address for an SF.REGR, SF.SWPI or SF.SWPQ function is no longer within the SCP field length.	Correct code error in SCP.	1AJ
SECURE FILES - CHECKPOINT ABORT.	Indicates a local file being checkpointed had secure file status set.	None.	CHKPT
SECURE MEMORY, DUMP DISABLED.	You either attempted to dump memory protected by the system, or entered a memory dump request after a protected command.	Refer to Security Control in NOS Reference Set, Volume 3, or user Field Length Dump Request Volume 4.	1AJ
SECURITY CONFLICT.	An attempted operation within the job would have resulted in a violation of security access levels or categories. The cause is described in the immediately preceding dayfile message.	Correct and retry.	1AJ
SECURITY VIOLATION ON FILE filename AT address.	One of the following: - You issued an OPEN request that specified an access level for a file that is not valid for your job. - You attempted to write on a direct access permanent file whose access level is lower than that of your job.	Set the access level of the file to the correct value and retry.	1MS
SFM ARGUMENT ERROR.	One of the following occurred: - The argument passed to SFM was out of range or the FET specified did not specify a buffer of at least 100B words. - The EST ordinal specified is undefined in the system.	Verify that SFM request is valid.	SFM
SFM BML MESSAGE LENGTH ERROR.	The specified BML message length is zero or greater than six. Legal values for length are one through six.	Specify a valid message length.	SFM

MESSAGE	SIGNIFICANCE	ACTION	ROUTINE
SFM - DAYFILE BUSY.	Action was requested on a busy dayfile.	Inform site analyst.	SFM
SFM DIRECT ACCESS FILE ERROR.	An error was encountered in the system sector of a direct access file.	Inform site analyst.	SFM
SFM DUPLICATE FILE FOUND.	The requested file is already attached to the control point.	Specify a unique file name.	SFM
SFM FILE NAME ERROR.	The file name, filename, passed through the FET is not a valid file name.	Specify a valid file name.	SFM
SFM FILE NOT ON MASS STORAGE.	The specified file does not reside on a mass storage device.	Specify a mass storage file.	SFM
SFM FNT FULL.	A fast attach file could not be created because the FNT was full.	Retry operation at a later time.	SFM
SFM I/O SEQUENCE ERROR.	Action has been requested on a busy file.	Wait until file is not busy and retry.	SFM
SFM INCORRECT DAYFILE CODE.	The dayfile code passed in the FET was not within range.	Inform site analyst.	SFM
SFM INCORRECT EQUIPMENT.	The specified equipment is not in the EST or is not mass storage.	Specify a valid mass storage equipment.	SFM
SFM INCORRECT FILE TYPE.	The file must be local for a system file and direct access for a fast attach file.	Specify a file of the correct file type.	SFM
SFM INCORRECT REQUEST.	One of the following occurred: - The requested function or origin type specified in the function call was not recognizable or SFM request was made and the auto recall bit was not set. - The request required an SSJ= entry point or subsystem ID, but none was present.	Verify that SFM request is valid.	SFM
SFM TRACK INTERLOCK ERROR.	Track was either interlocked when it should not have been or not interlocked when it should have been.	Contact Central Software Support.	SFM
SFM UNABLE TO INTERLOCK DEVICE.	SFM request was not performed because the selected device could not be interlocked.	Contact Central Software Support.	SFM
SFM - UNCORRECTABLE RMS ERROR.	An uncorrectable RMS error has been detected during an I/O operation.	Contact Central Software Support.	SFM
jsn SHARE TABLE MISMATCH.	Dayfile message indicating that RESEX internal problem occurred. While processing the job with the specified job sequence name.	Inform site analyst.	RESEX
SI CODED FORMAT INCORRECT.	An attempt was made to perform a read or write operation on an SI tape with the coded bit set in the FET.	Resubmit the job with binary specified for the operation on the SI tape.	1MT
SL NOT VALIDATED.	The SRU limit requested exceeds that for which you are authorized.	Request smaller SRU limit.	CPM
STACK PURGING NOT DESELECTABLE.	You attempted to change the purging bit with the EREXIT function. This cannot be done on 6000 or CYBER 70 systems.	None.	CPM
pfn STAGE INITIATED.	The file does not reside on disk. It is currently being staged from alternate storage to disk.	Wait until stage operation is complete.	PFM
STATUS ERROR, filename AT address.	An error was encountered during magnetic tape processing. A second message line describes the error in more detail.	Retry or inform site analyst.	1MT
STEP CONDITION.	Step mode flag set in the PSD register caused the program to interrupt at the end of a program instruction with an exchange jump to EEA (the error exit address in the exchange package).	Inform site analyst.	1AJ
SUBSYSTEM ABORTED.	Your job was connected (either long term connection or wait response set) to a subsystem which aborted.	Retry later.	1AJ
SYSTEM ABORT.	Possible causes include the following. - Incorrect USER command. - Attempt to access a restricted subsystem. - Operator evicted job. - Unrecognizable error flag. - SSJ= block outside field length. - 1RI detected a bad rollout file. - 1RO detected an unrecoverable extended memory parity error during rollout of a job's extended memory field length.	If the cause was an incorrect USER command or an attempt to access a restricted subsystem, correct the job and rerun. Otherwise, inform a site analyst.	1AJ

MESSAGE	SIGNIFICANCE	ACTION	ROUTINE
SYSTEM CHECKPOINT ABORT.	A subsystem has aborted due to a CHECK POINT SYSTEM request initiated by the operator.	None.	1CK
SYSTEM ERROR.	A software or hardware system error occurred. This message follows a more specific message in the dayfile.	Refer to action for the associated message.	MODVAL PFM
SYSTEM ERROR.	LFM cannot complete the requested LFM function because the calling program has a DMP= entry point.	Inform site analyst.	BLANK RESEX
SYSTEM ERROR.	Requested function cannot be completed because the calling program has a DMP= entry point.	Contact Central Software Support.	LFM
SYSTEM SECTOR ERROR.	The system sector of an indirect access permanent file contains an error (error log and dayfile message). This indicates that the file has been destroyed.	Inform site analyst. If many of these errors occur, the site analyst should perform a full PFDUMP, total INITIALIZE, and full PFLOAD on the device.	PFM
TAPE FORMAT PROBABLY WRONG.	This message is issued in addition to one of the following messages: - BLOCK SEQUENCE ERROR, filename AT addr. - BLOCK TOO LARGE, filename AT addr. - WRONG PARITY, filename AT addr.	Ensure accuracy of format (F) parameter on command or macro.	1MT
TIME LIMIT.	The execution time limit for a job step expired resulting in job termination.	If a time limit was set for the job, include a SETTL command requesting a longer time limit for the job step. If the job step time limit was the maximum for which you are validated, request a larger time limit or decrease the amount of processing to be performed by the job step.	1AJ
TL NOT VALIDATED.	The time limit requested exceeds that for which you are authorized.	Request smaller time limit.	CPM
TLX - ARGUMENT ERROR.	The specified function parameter address is out of range.	Specify a valid parameter address.	TLX
TOO MANY PERMANENT FILES.	The number of files in your catalog exceeds your limit.	Purge one or more permanent files to allow you to save or define additional files.	PFM
TOO MUCH INDIRECT ACCESS FILE SPACE.	The cumulative size of the indirect access files in your catalog exceeds your limit.	Purge or shorten one or more indirect access files to allow additional permanent file space.	PFM
TOTAL ASSIGNED COUNT ERROR.	Dayfile message indicating RESEX internal problem (sum of individual resource assigned counts differs from total assigned count in demand file entry).	Inform site analyst.	RESEX
TOTAL DEMAND COUNT ERROR.	Dayfile message indicating RESEX internal problem occurred (sum of individual resource demand counts differs from total demand count in demand file entry).	Inform site analyst.	RESEX
TRACK LIMIT.	No allocatable tracks remain on your permanent file equipment (error log and dayfile message).	None; job will continue as tracks become available. If problem persists, contact central software support.	PFM
TRACK LIMIT, LVLX.	1MS is waiting for temporary file mass storage space on any mass storage device with access level LVLX.	Inform site analyst.	1MS
TRAILER BLOCK COUNT ERROR, filename AT address.	The block count in the EOF1 or EOV1 label did not match the block count maintained by the tape executive during the read operation.	Inform site analyst.	1MT

MESSAGE	SIGNIFICANCE	ACTION	ROUTINE
UNIDENTIFIED PROGRAM FORMAT.	The file you requested to be loaded was in an unrecognizable format.	Check the format of the file.	1AJ
UNIT HUNG UP ON EOP OR BUSY, filename AT address.	Unit did not receive EOP on unit busy.	Inform site analyst.	1MT
UNNECESSARY CIO FCT. nnn ON filename AT address.	The read or write CIO function specified by nnn was unnecessary, since your I/O buffer was already full (read) or empty (write) and no data could be transferred for this CIO call. The message is issued to the job dayfile only if analyst logging is enabled on the system (console entry).	Ignore or correct program to be more efficient.	1MS
UNRECOGNIZED TERMINAL MODEL.	You entered an invalid terminal model mnemonic as a parameter in the SCREEN or LINE command.	Retry with valid mnemonic.	CONTROL
USER ACCESS NOT VALID.	You tried to perform an operation for which you are not authorized. Possible causes include attempts to access a file or equipment which you are not authorized to access.	Ensure accuracy of command or determine proper validation requirements via LIMITS command.	LFM
USER ACCESS NOT VALID.	You tried to perform an operation for which you are not authorized. Possible causes include attempts to - Run a system origin job from nonsystem origin. - Access a restricted subsystem without proper validation. - Enter an incorrect SRU value. - Use the V carriage control character without validation.	Ensure accuracy of command or macro, or determine proper validation requirements via LIMITS command.	DSD EXUBUT EXCSLV MSI QFSP RESEX 1MA IAFEX
USER ACCESS NOT VALID.	You are not authorized to create direct access or indirect access files or to access auxiliary devices.	Contact site personnel concerning validations.	PFM
USER ACCESS NOT VALID.	The user name or password could not be validated, or a secondary USER command was encountered while secondary USER commands were disabled.	Verify that user name and password are valid. If secondary USER commands are disabled, ensure that no secondary USER commands are present.	CPM
USER ACCESS NOT VALID.	The SRU or time limit request is outside of your validated range.	Retry with a lower value.	CONTROL
USER SECURITY COUNT EXHAUSTED.	The security count for the user name specified has been decremented to zero. You are denied all access to the operating system until the security administrator resets your security count.	Contact site personnel to reestablish access.	CPM IAFEX 1AJ
VEJ - BUFFER ARGUMENT ERROR.	Dayfile message indicating that FET buffer pointers are incorrect. (FWA<LWA<FL) was not true or TID (terminal id) with complement address was not within the field length.	Write a PSR and include support materials to allow CDC to duplicate the problem.	VEJ
VEJ - INCORRECT REQUEST.	Dayfile message indicating that one of the following conditions has occurred: - VEJ was not called by a subsystem. - The FET address was out of range.	Write a PSR and include support materials to allow CDC to duplicate the problem.	VEJ
VSN FILE ERROR.	Dayfile message indicating that a RESEX internal problem occurred (VSN file entry does not match job identification).	Inform site analyst.	RESEX
pfn WAITING ALTERNATE STORAGE EXEC.	The file must be staged from alternate storage to disk and the alternate storage executive is not currently available to perform the stage operation.	Wait for executive to be started or inform site operator that the alternate storage executive should be ilitialized.	PFM
WAITING FOR MAGNET.	The job is waiting for the magnetic tape subsystem to be activated.	Wait for the operator to activate MAGNET or terminate job.	RESEX
WAITING FOR NFL.	Informative message stating that your job has been delayed and/or rolled out waiting for NFL space. The job has not been aborted; it will eventually continue. The message is visible via ENQUIRE.	None.	PFM
WAITING FOR PN=packname, type.	The job is waiting for the operator to mount pack packname on device type type.	Wait until the operator mounts the requested pack or terminate job.	RESEX

MESSAGE	SIGNIFICANCE	ACTION	ROUTINE
WAITING FOR RESOURCE FILE.	The job is waiting for the resource demand file or VSN file to become available.	Wait until resource file becomes available or terminate job. To operator: If job is not rolled out and this message persists, inform site analyst or drop the job. If the operator decides to override an interrupted job at this point, the preview data in the demand file is not cleared and the E,P display continues to show the VSN request associated with the job until you log off or issue a subsequent request for tape or pack.	RESEX
WAITING FOR RESOURCES.	The job is waiting for sufficient resources to allow assignment of the tape/pack without causing a system deadlock.	Wait until the resources become available or terminate job.	RESEX
WAITING FOR VSN= vsn, type.	The job is waiting for the operator to mount the tape with VSN vsn on the specified type (MT, HD, PE, or GE). VSN= SCRATCH indicates that any scratch tape is acceptable.	Wait for the operator to mount the tape or terminate the job.	RESEX
WAITING ON TRACK LIMIT.	The job is waiting for additional tracks on the familyname device containing the resource demand and VSN files.	Wait for the additional tracks or terminate job.	RESEX
WRITE-DOWN OF DATA PROHIBITED.	One of the following: - On an APPEND or REPLACE command or macro, the access level of the local file is higher than the access level of the permanent file. - On a DEFINE command or macro, the access level of a local file having the same name as the file specified has a lower access level than the job.	Retry using a higher access level for the permanent file (APPEND or REPLACE), or change the job's access level and then retry (DEFINE).	PFM
WRITE-DOWN PROHIBITED ON FILE, filename AT address.	The access level of your job is higher than the access level of file filename, and the file is a tape file or a direct access file.	None. You may not write on this file while your job is at the current access level.	1MS
WRITE ON READ-ONLY FILE filename AT address.	Either you attempted to write on a file with write lock-out, or the direct access file was not attached in write mode.	Reattach file in write mode or clear write interlock.	1MS IAFEX
WRITE OVER LABEL ILLEGAL.	You attempted to write over the VOL1 label.	Have the operator blank label the tape.	1MT
WRITEDOWN PROHIBITED ON FILE FILENAME AT address.	The access level of your job is higher than the access level of file filename, and the file is a tape file or a direct access permanent file.	None. You may not write on this file while your job is at the current access level.	1MS
WRONG PARITY, filename AT address.	A seven-track tape is being read in opposite parity from which it was written.	Ensure accuracy of format parameter (F) on command or macro.	1MT
XD/XT EXCEEDS MAXIMUM.	The XD parameter for a password or file permission is more than the maximum number of days past the current date, or the XT parameter is larger than the maximum allowed.	Correct XD or XT parameter and retry.	PFM
XL BUFFER/FET PARAMETER ERROR, filename AT address.	One of the following: - HDR1 label in extended label buffer or FET contains a nonnumeric display code value in a numeric field. - Character count in header word preceding labels in the extended label buffer does not equal 80.	Correct condition that caused error and retry.	1MT
200 BPI WRITE INCORRECT.	The tape unit (667 or 677) cannot record data at 200 bpi.	Specify a different tape density.	1MT

GLOSSARY C

Abort

To terminate a program, job, or job step when an error condition (hardware or software) exists from which the program or computer cannot recover.

Absolute Assembly

In the context of this manual, a COMPASS program assembled with the ABS pseudo instruction in which there are no references to external entry points. Compare with Relocatable Assembly.

Access Category

Refer to File Access Category, Job Access Category Set, and System Access Category Set.

Access Level

A property of each file, job, and equipment on a secured system that is used to indicate the sensitivity of information in the file or job, or the sensitivity of information that can be processed by the equipment. On a secured system, there are up to eight access levels corresponding to increasing levels of sensitivity, and each user is authorized to access some or all of those levels. Refer also to Equipment Access Levels, File Access Level, Job Access Level, and System Access Levels.

Access Level Limits

Refer to Job Access Level Limits.

Account Block

A string of commands between two CHARGE commands or between a CHARGE command and end-of-job.

Allocatable Device

A storage device allocated by the system without operator intervention that can be shared by more than one job.

Alphanumeric

Consisting of alphabetic and/or numeric characters only.

Alternate User Name

A user name specified in a permanent file request which indicates the action is to be taken on an alternate user's permanent file.

ANSI

American National Standards Institute. An organization that establishes standards for the benefit of its member organizations.

ASCII

American National Standard Code for Information Interchange. The standard character set and code used for information interchange between systems.

Auto Recall

The act of a program releasing control of the CPU until a requested function is complete. Refer to Recall.

Auxiliary Device

Mass storage device that is not part of a permanent file family. Auxiliary devices can contain direct or indirect access permanent files.

Block

A grouping of user records created for efficiency in transfer between memory and storage devices. For magnetic tapes, it is the information between interrecord gaps. Block size is specified with the LABEL macro for L format tapes.

BOI

Refer to Beginning-of-Information.

Buffer

An intermediate storage area used to compensate for a difference in rates of data flow, or times of event occurrence, when transmitting data between central memory and an external device during input/output operations.

Byte

A group of 12 bits. Five bytes comprise a 60-bit central memory word. Bytes are numbered 0 through 4 from the left.

Central Memory (CM)

The main storage device whose storage cells (words) can be addressed by a computer program and from which instructions and data can be loaded directly into registers from which the instructions can be executed or from which data can be manipulated.

Central Processor Unit (CPU)

The high-speed arithmetic unit that performs the addition, subtraction, multiplication, division, incrementing, logical operations, and branching instructions needed to execute programs.

Checkpoint

The process of writing to a magnetic tape or mass storage file a copy of your job's central memory, the system information used for job control, and the names and contents of all assigned files that are identified in a CHECKPT request.

Checkpoint File

File on which the results of a partially completed job are dumped when a checkpoint request is processed.

Checksum

A numeric value used to verify a file.

Circular Buffer

A temporary central memory storage area that contains data during input/output operations. Routines that process I/O treat the first word of the buffer area as contiguous to the last word of the buffer area.

Command Record

The first, and possibly only, record of a job file consisting of command images that start with the Job command and end with the first EOR, EOF, or EOI. Also refers to a procedure containing commands.

Common Deck

A subprogram or group of macro or symbol definitions that are accessed from a program library using the Modify CALL directive or COMPASS XTEXT pseudoinstruction.

Compare/Move Unit (CMU)

The hardware that executes the CPU instructions for moving and comparing data fields consisting of strings of 6-bit characters.

COMPASS

The standard assembly language used with CYBER 170, CYBER 70, and 6000 Computer Systems. Also, the command used to assemble a program written in the COMPASS assembly language.

Control Byte

A 12-bit code that changes the current input or output mode at an interactive terminal.

Control Point

The portion of central memory that is assigned to a job. When a job is allocated a portion of central memory, it becomes eligible for assignment to the central processor for execution.

CPU

Refer to Central Processor Unit.

CYBER Loader

The utility that prepares programs for execution by placing program instruction and data blocks in central memory.

CYBER Record Manager (CRM)

A software product that allows a variety of record types, blocking types, and file organizations to be created and accessed. Products like COBOL 5, FORTRAN Extended 4, FORTRAN 5, Sort/Merge 4, Sort/Merge 5, ALGOL 5, and DMS 170 use CRM to manage execution time input/output. Neither the input/output of the operating system nor that of most of the system utilities such as COPY or SKIPF is implemented through CYBER Record Manager. All CYBER Record Manager file processing requests ultimately pass through the operating system input/output routines.

Dayfile

A chronological file created during job execution which forms a permanent accounting and job history record. Dayfile messages are generated by operator action or when commands are processed. A copy of the dayfile is printed with the output for each job. You must explicitly request it in an interactive job.

Default

A system-supplied option used when you do not supply the option.

Device Type Code

The 12-bit display code of the type of device upon which a file resides.

Direct Access File

A NOS permanent mass storage file that can be attached to the your job. All changes to this file are made on the file itself rather than a temporary copy of the file (compare with Indirect Access File).

Display Code

A 6-bit character code set used to represent alphanumeric and special characters.

Disposition Code

A two-character mnemonic indicating destination queue and format for processing a file named on a ROUTE function.

ECS

Extended Core Storage. Refer to Extended Memory.

Empty PRU/Record

A PRU that contains no user data. Refer also to Zero Length PRU.

End-of-File (EOF)

A boundary within a sequential file, but not necessarily the end of a file that can be referenced by name. The actual end of a named file is defined by EOI. For labeled tape, EOF and EOI (denoted by the EOF1 label) are the same. For multifile tape files, EOF and EOI do not correspond. In the product set manuals, an end-of-file is also referred to as an end-of-partition.

End-of-Information (EOI)

The end of data on a file. Information appearing after this point is not considered part of file data. In card decks, a card with a 6/7/8/9 multiple punch in column 1. On mass storage devices, the position of the last written data. On labeled tape, it is the EOF1 label. CYBER Record Manager defines end-of-information in terms of file residency and organization.

End-of-Line (EOL)

A separator in coded files that is represented by 12 bits of zero, right-justified in a 60-bit word. In some cases, up to 66 bits of trailing zero bits are considered an end-of-line.

End-of-Record (EOR)

The terminator of a logical record. On a PRU device, a short PRU or a zero length PRU with a level designator of 0 indicates EOR. On tapes that are not PRU devices, an interrecord gap indicates EOR.

End-of-Tape (EOT)

A reflective strip near the end of a magnetic tape. It is used to signal termination of operations on a particular tape volume. At least 5.5 metres (18 feet) of tape must follow this marker.

Entry Point

A location within a program or procedure that can be referenced from other programs. Each entry point has a unique name with which it is associated. Refer to External Reference.

EOF

Refer to End-of-File.

EOI

Refer to End-of-Information.

EOL

Refer to End-of-Line.

EOR

Refer to End-of-Record.

EOT

Refer to End-of-Tape.

Equipment Access Levels

A range of access levels specified for each equipment on a secured system. In order for a file to be stored or output on a given equipment, the file's access level must be within the equipment access levels for that equipment.

Equipment Code

Refer to Device Type Code.

Equipment Status Table (EST)

A list of all peripheral devices connected to the system. Each table entry indicates the status of a particular device.

EST Ordinal

The number designating the position of an entry within the equipment status table (EST) established at each installation. Devices are identified in operator commands by EST ordinals.

Exchange Jump

Execution of a CPU program is initiated by an exchange jump. The program is defined by the contents of the exchange package area before the exchange jump took place. For the program to execute, the proper contents of its operational registers must be loaded into the CPU. These register contents are what is contained in the exchange package area associated with the program.

Exchange Package

A 20_8-word table containing information used in exchange jumps during job execution: contents of central processor registers, RA and FL in central memory and in extended memory, and the program address. The table is stored in the job's field length and can be printed as part of the output of a central memory dump.

Exit Mode

A group of flags that specify the types of errors that can cause the CPU to abort.

Extended Core Storage (ECS)

A type of extended memory that is an option available for 6000 Computer Systems, CYBER 70 Computer Systems, and CYBER 170 Computer Systems, except for models 176 and CYBER 180-class models. Refer to Extended Memory.

Extended Labeled Processing

A tape processing mode in which all tape labels (including optional labels) are read into a label buffer for further processing.

Extended Memory

An additional portion of memory available as an option. This memory can be used for program and data storage, but not for program execution. Special hardware instructions exist for transferring data between central memory and extended memory. Extended memory consists of either extended core storage (ECS), large-core memory extended (LCME), extended semiconductor memory (ESM), or unified extended memory (UEM).

Extended Semiconductor Memory (ESM)

A type of extended memory that is an option available for 6000 Computer Systems, CYBER 70 Computer Systems, and CYBER 170 Computer Systems, except for models 176 and CYBER 180-class models. Refer to Extended Memory.

External Reference

A reference in a program to an entry point in another program. Throughout the loading process, externals are matched to entry point (this is also referred to as satisfying externals); that is, addresses referencing externals are supplied with the correct address.

Family Device

Mass storage permanent file device associated with a specific system. A family may consist of from 1 through 63 logical devices. Normally, a system runs with one family of permanent file devices available. However, additional families may be introduced during normal operation. This enables users associated with the additional families to access their permanent files via the alternate family.

Fast Dynamic Loader (FDL)

A facility that provides fast loading and unloading of specially formatted code called capsules. The amount of memory required for job execution can be greatly reduced because capsules can be easily loaded and unloaded as needed, freeing memory for other uses.

FET

Refer to File Environment Table.

Field Length

The area in central memory allocated to a particular job; the only part of central memory that a job can directly access. Also the number of central memory words required to process a job.

Field Length Extended (FLE)

Amount of extended memory assigned to an executing job.

File

1. A set of information that begins at beginning-of-information (BOI), ends at end-of-information (EOI), and is referenced by a local file name.

2. That portion of a multifile file terminated by an end-of-file (EOF).

3. Data recorded on a magnetic tape beginning after an HDR1 label and ending before an EOF1 label.

NOS commands requiring a parameter that is a file name refer to definition 1. Commands requiring a parameter that specifies the number of files refer to definition 2. Definition 3 applies only to labeled magnetic tapes.

File Access Category

A property of a permanent file used by the creator of the file on a secured system to restrict access of the file to a particular group of users. A secured system supports up to 32 access categories, and each user is authorized to use some, all, or none of those categories. Refer also to Job Access Category Set and System Access Category Set.

File Access Level

A property of each file on a secured system used to indicate the sensitivity of information contained on the file. A file is assigned the current job access level by default when it is created or stored; the file creator may specify any access level for that file that is within the set of access levels valid for the job, the system, the file creator, and (for interactive jobs) the communication line to the host mainframe. Any user who accesses a file on a secured system must be validated for the access level of the file. Refer also to Access Level, Job Access Level, and Job Access Level Limits.

File Environment Table (FET)

A table within a progam's field length that defines the current status and properties of a file being used by the program. The program communicates with operating system input/output routines through the FET. One FET exists for each file in use by the program.

File Flushing

The process of writing the contents of a file's circular buffer to mass storage when certain conditions are met.

File Name Table (FNT)

A system-managed table that contains job control information for all active files in the system. Files that are local to a job have an entry in the job's local file name table (refer to Local File Name Table); system files have an entry in the system file name table (refer to System File Name Table). Each file name table entry consists of a file name table word, a file status table word, and, for local files, a file utility table word.

File Name Table (FNT) Word

The first word of every file name table entry. This word contains the local file name, file type, and other job control information.

File Set

One or more tape files referred to by the set identifier on a tape assignment command or macro. A file set may consist of:

1. One file recorded on a single volume.
2. More than one file recorded on a single volume.
3. One file recorded on more than one volume.
4. More than one file recorded on more than one volume.

All files within a file set have the same set identifier in their HDR1 labels.

File Status Table (FST) Word

The second word of every file name table entry. This word contains information pertaining to the file's location on mass storage and other job control information.

File Utility Table (FUT) Word

The third word of every file name table entry. This word contains the 12-bit installation area for the file; the remainder of the word is reserved for future use.

FL

Refer to Field Length.

Flag

A character or bit that signals the occurrence or presence of a particular condition.

FLE

Refer to Field Length Extended.

FNT

Refer to Local File Name Table.

FNT Word

Refer to File Name Table Word.

Foreign Tape

NOS classifies all non-CDC standard tapes into either of two tape groups, foreign or stranger. The foreign tapes are defined by the F data format, which has an unknown maximum block size. Contrast with Stranger Tape.

Frame

A tape recording unit made up of 1 bit from each tape track (7 bits for 7-track tape and 9 bits for 9-track tape). Each frame on a coded tape usually represents one character.

FST Word

Refer to File Status Table Word.

Full Track (FT)

Reading/writing sequential sectors on a rotating mass storage device.

Function Processor

A system CPU or PP program that the user can call by placing a request in location RA+1. Function processors perform input/output, local and permanent file manipulations, and so on.

Generation

The position of a file within a series of files, each file developed from the preceding file. The generation number and generation version number of a tape file can be entered in its HDR1 label.

Global Library Set

An ordered set of libraries used for program loads and for satisfying externals; these libraries remain in effect throughout job execution unless specifically changed.

Half Track (HT)

Reading/writing alternate sectors on an 844 or 885 disk pack.

Indirect Access File

A NOS permanent file that is accessed by making a temporary copy of the file (GET macro). It is created or altered by saving or replacing the contents of an existing working file (REPLACE or SAVE macro).

Input File Type

Job file. Its first record is a control statement record, which may be followed by records containing data, directives, or programs used by job steps.

Interactive Programs

CPU programs in which the user and the computer communicate with each other.

Interrecord Gap

Space skipped between the writing of data blocks on magnetic tape.

Interrupt

To stop a running program in such a way that it can be resumed at a later time. Also a special control signal which, when issued, causes action as described by the program.

Job Access Category Set

On a secured system, a set of access categories is set when the job is initiated. This set is the intersection of the user's set of validated access categories and the system access category set. Refer also to File Access Category and System Access Category Set.

Job Access Level

On a secured system, each job has an access level. This is the default access level that is assigned to files that are created or stored in the job. A job's initial access level is the lower access level limit for the job. The job's access level is automatically raised to the access level of any file from which information is read. The job access level can also be changed by the user. Refer also to Job Access Level Limits.

Job Access Level Limits

An upper limit and a lower limit that determine the range of access levels that are valid for a particular job on a secured system. All files used in a given job must have an access level within the job's access level limits.

Job Step

An individual command, procedure or loader sequence. A group of job steps forms a job stream, command record, or procedure file.

Label

A block at the beginning or end of a magnetic tape volume or file, which serves to identify and/or delimit that volume or file.

Large Central Memory Extended (LCME)

A type of extended memory that is an option available for model 176. Refer to Extended Memory.

LCME

Refer to Large Central Memory Extended.

Level Designator

Refer to Level Number.

Level Number

The level number is an octal number in the terminating marker of a PRU, ranging from 00 to 17 (octal). A level 17 in an empty PRU designates an EOF. A level 0 in a short PRU designates an EOR. A level designator of 1 is used as a flag to indicate that a short PRU of data is actually a line of input data from a terminal. The level designator is returned to the FET for some read functions and to the trailing control word for a READCW function.

lfn

Refer to Local File Name.

Library

A file or collection of files containing executable programs and tables needed to locate and load the programs. A system library can contain peripheral processor programs in addition to the central processor programs. A user library is file formatted as a library but is not available to a job until it has been explicitly declared via CYBER Loader commands or macros.

Library File

Either a read-only file that can be accessed by several users simultaneously or a file you specify on a LIBRARY command.

Line

Refer to Zero-Byte Terminator and End-of-Line.

List of Files

A table containing names of files that are to be considered for file flushing.

Load Point

Metallic strip marking the beginning of the recordable portion of a magnetic tape. Data, including labels, is written after the load point. A rewind positions a single file volume to the load point.

Local File

Any file that is currently associated with a job. Local files include all temporary files and attached direct access files.

Local File Name

The file name assigned to a file while it is local (assigned) to a job. The name is contained in the local file name table.

Local File Name Table (FNT)

A system-managed table that contains the local file name, the file type, and other job control information.

Locked File

A file on which you cannot write.

Logical Identifier (LID)

A three-character alphanumeric string used to identify a particular mainframe in a loosely coupled network. LIDs are identified by your site.

Logical Record

A data grouping that consists of zero or more PRUs and ends with a short PRU or a zero-length PRU.

Macro

A sequence of source statements that are saved and then assembled whenever needed through a macro call.

MLRS

Maximum logical record size. A number that specifies the maximum size for a logical record for S and L tape formats.

Multifile File

A file containing more than one logical file. It begins at BOI and ends at EOI. On a labeled tape, a multifile file is delimited by corresponding HDR1 and EOF1 labels.

Multifile Set

A tape file set having more than one tape file.

Network

A data and message switching and routing system used to provide communication between terminals, applications, and mainframes.

Network Terminal

A terminal that communicates with the operating system through the network.

Nonallocatable Device

A device (such as a magnetic tape unit, card reader, card punch, or line printer) which can be used only by one job at a time.

Nonstandard Tape Label

A tape label format for a tape whose data cannot be processed by the system because the label is not one of the supported label types. If the tape is at load point, a subsequent read operation skips to the first tape mark.

Old Program Library (OPL)

A Modify-formatted program library that contains source code for system routines and common decks.

opdef

A sequence of source statements that are saved and then assembled whenever needed through an opdef call. Differs from a macro in that the assembler interprets the call by examining the format or syntax of the instruction rather than the contents of the operation field alone.

OPL

Refer to Old Program Library.

Overlay

One or more relocatable program modules that have been relocated and linked into a single absolute program. It can be a main, primary, or secondary overlay.

Pack Name

A one- through seven-character name that identifies the auxiliary device to be accessed in a permanent file request.

Parameter

A variable that is given a specific value for a particular purpose or process.

Parity

In writing data, an extra bit is either set or cleared in each byte so that every byte has either an odd number of set bits (odd parity) or an even number of set bits (even parity). Parity is checked on a read for error detection and possible recovery.

Password

1. A system access word that must be used in addition to the user name at login.

2. A file access word that controls access to a particular file by alternate users.

Permanent File

A mass storage file that is cataloged by the system so that its location and identification are always known to the system. Permanent files cannot be destroyed accidentally during normal system operation. They are protected by the system from unauthorized access according to privacy controls specified when they are created.

Permanent File Catalog (PFC)

A 16-word entry that the system maintains and uses to determine attributes of a permanent file.

Permanent File Family

Permanent files which reside on the family devices of a specific system.

Physical Record

On magnetic tape, information between interrecord gaps (refer to Block). It need not contain a fixed amount of data. For mass storage, refer to Physical Record Unit.

Physical Record Unit (PRU)

The amount of information transmitted by a single physical operation of a specified device. For mass storage files, a PRU is 64 central memory words (640 characters); for magnetic tape files, the size of the PRU depends upon the tape format. A PRU that is not full of user data is called a short PRU; a PRU that has a level terminator but no user data is called a zero-length PRU.

PP

Refer to Peripheral Processor.

Primary File

A temporary file created with either the OLD, NEW, LIB, or PRIMARY command or the PRIMARY macro for most systems. The primary file is assumed to be the file on which most operations are performed unless another file is specified. There can be only one primary file associated with a job. The primary file is rewound before each job step given.

Print File

An output file containing data to be printed at a central site or remote batch line printer.

Private Auxiliary Device

Auxiliary device associated with a specific user name. Only that user name may create files on the device, although other users may be permitted to access files which reside on the device.

Program Library

The file generated by Modify that contains the decks of line images. Line images in the program library are in a format that can be manipulated by Modify.

PRU

Refer to Physical Record Unit.

PRU Device

A magnetic tape file or a mass storage device. Records on these devices are written in physical record units (PRUs).

Pseudoinstruction

An assembler-defined instruction appearing in the operation field of a statement. It normally does not specify the assembly of a single machine instruction, but instead specifies some other assembly process (such as symbol definition or listing control).

Public Auxiliary Device

Auxiliary device available for access by all validated users knowing the correct pack name. Additional validation is required to create or replace files on an auxiliary device.

Punch File

Output file containing data to be punched on cards.

QFT

Refer to Queued File Table.

Qualified Symbol

A symbol defined when a qualifier is in effect during assembly. Through qualification, the same symbol can be referred to in different subprograms without conflict.

Queued File Table (QFT)

A central memory resident table containing a fourword entry for all active input and output queue files.

Random Access

Access method by which any record in a file can be accessed at any time. Applies only to mass storage files with an organization other than sequential. Compare with Sequential Access.

Random File

A file with an index entry to each record in the file. A file on a rotating mass storage device is a randomfile only when the random bit is set in the file environment table. The last record of the file is an index.

Recall

The state of a program when it has released control of the central processor until a fixed time has elapsed (periodic recall) or until a requested function is completed (auto recall). Recall is a system action request as well as an optional parameter of some file action requests.

Record

Refer to Logical Record.

Record Separator

In NOS, another name for an end-of-record (EOR).

Reel

Refer to Volume.

Reference Address (RA and RAE)

RA is the absolute central memory address that is the starting or zero relative address assigned to a program. Addresses within the program are relative to RA. RA+1 is used as the communication word between the user program and the system. RAE is the absolute extended memory starting address assigned to a program.

Register

A register is a storage device used to hold binary data. There are 24 hardware registers (eight A registers, eight B registers, and eight X registers) in the CPU that can be directly controlled by a COMPASS program. The A registers contain addresses of words in central memory, the B registers are used for incrementing and indexing, and the X registers contain operands used in calculations and the results of these calculations. Other hardware registers contain information that is used by the operating system or by system hardware.

Relative Address

All addresses in a relocatable program are relative to a base address of zero. When a relocatable program is loaded for execution, the zero base address is assigned to a reference address. At that time, all addresses in the program become relative to the reference address.

Relocatable Assembly

An assembled program that contains references to external entry points. Compare with Absolute Assembly.

Removable Device

A rotating mass storage device that the system can make unavailable. It is not necessary that the device be physically detachable from the rotating mass storage drive.

Rollout

The removal of jobs from central memory to mass storage before execution is complete, so the control point and central memory can be assigned to another job. A job is rolled out when it is waiting for an external event, when its control point and/or central memory is needed by a higher priority job, or when it exceeds its central memory time slice.

Rollout File

A file containing a job (and system information) that has been temporarily removed from the main processing area of the system.

Rotating Mass Storage (RMS)

A disk storage device.

Sector

Refer to Physical Record Unit.

Secured System

A system in which a mandatory security mechanism has been enabled during deadstart. A secured system protects information by enforcing restrictions based on access levels and access categories, and restricts many sensitive system functions to security administrators.

Security Administrator

A secured system prevents users and operators from performing certain functions that could result in the unauthorized disclosure of information. These functions can only be performed by a person who is designated a security administrator. A security administrator is always authorized to access the highest level of information stored on the system. This person performs functions in the areas of installation, user validation, system operation, and system maintenance.

Security Count

The number of security violations you have left before you are denied access to the system.

Security Unlock Status

This status of the system console applies only to a secured system and must be set by a security adminstrator. The console must be in security unlock status in order for the security administrator to perform certain functions that are restricted on a secured system.

Sense Switches

Six 1-bit flags located in the user's control point area and in bits 11 through 6 of RA+0 of the job field length. These bits can be used to communicate with system routines (refer to the ONSW and OFFSW macros in section 6).

Sequential Access

A method in which only the record located at the current file position can be accessed. Refer to Random Access.

Sequential (SQ) File

A file in which records are accessed in the order in which they occur. Any file can be accessed sequentially.

Short PRU

A PRU that does not contain the maximum amount of character data allowed for a PRU. Refer to Zero-Length PRU.

Special Entry Point

An entry point that enables system programs to perform special functions.

SRU

Refer to System Resource Unit.

Standard Labeled Tape

A tape with labels conforming to American National Standard Magnetic Tape Labels for Information Interchange X3.27-1969. Also called a system labeled tape.

Stranger Tape

NOS classifies all non-CDC standard tapes into either of two tape groups, foreign or stranger. The stranger tapes are defined by the S data format, which has a maximum block size of 1000_8 CM words. Contrast with Foreign Tape.

Symbol

A set of characters that identifies a value and its associated attributes.

System Access Category Set

On a secured system, a set of access categories are set during level 0 deadstart. This set may consist of some, all, or none of the 32 possible access categories. While the system is running, users may only use access categories that are within the set of system access categories. Refer also to File Access Category and Job Access Category Set.

System Access Levels

On a secured system, a range of access levels is set during level 0 deadstart. This range may contain some or all of the eight possible access levels. While the system is running, users may only use access levels that are within the range of system access levels.

System File

A file that can be accessed only by a system program.

System File Name Table

A system-managed table that contains a file name table entry for the various system files.

System Labeled Tape

Refer to Standard Labeled Tape.

System Library (SYSLIB)

The collection of tables and object language programs that reside in central memory or on mass storage and are necessary for running the operating system and its product set.

System Request

A request placed in location RA+1 for a function processor to perform a special process.

System Resource Unit (SRU)

A unit of measurement of system usage. The number of SRUs includes the central processor time, memory usage, and input/output resources used for a given job.

System Text

A set of tables containing symbol, micro, macro, and opdef definitions that can be saved on a file to be accessed by other programs.

Tape Format

A parameter that specifies the internal recording format of a magnetic tape.

Tape Mark

A delimiter written on tapes under operating system control to separate label groups, files, and/or labels. Interpretation depends on the tape format.

Temporary File

A file that is currently associated with a job and is not a permanent file. Temporary files cease to exist when they are returned to the system (either by means of a command or macro or upon job termination).

Time Slice

The amount of CPU or CM time a job is allowed to use before the system lowers its priority to allow other jobs to execute.

Timed/Event Rollout

A condition in which an executing job has been temporarily removed from central memory but will be rolled back into central memory when a specified event (such as a file is no longer busy) or a specified time period has elapsed.

UEM

Refer to Unified Extended Memory.

Unified Extended Memory

A type of extended memory that is available as an option for CYBER 180-class models and models 865 and 875. UEM differs from other types of extended memory in that it is a portion of central memory and not a separate memory unit. Refer to Extended Memory.

Unlabeled Tape

A magnetic tape that does not have a header label. Unlabeled tapes generated by the operating system contain a trailer label similar to the trailer for a standard labeled tape.

Unsecured System

A system in which the security mechanism has not been enabled during deadstart. The restrictions based on access levels and access categories are not enforced on an unsecured system.

User Index

A unique 17-bit identifier that is associated with each user name. The user index is used by the permanent file manager to identify the device and catalog track for the user's permanent files.

Validation File

File containing validation information for all users (user names, passwords, resources allowed, and so on).

Volume

A reel of magnetic tape. A given file can be composed of more than one volume.

Volume Serial Number (VSN)

A one- through six-character identifier that identifies the volume of magnetic tape to the system.

VSN

Refer to Volume Serial Number.

Word

A group of bits that occupy a single memory cell. A central memory or extended memory word is 60 bits in length. The bits are numbered 59 through 0 starting from the left. A word is also composed of five 12-bit bytes, numbered 0 through 4 from the left.

Write Ring

A circular device inserted into a tape reel indicating to the tape unit that it can write on that reel. NOS checks for the presence of a write ring if you request it.

Zero-Byte Terminator

The 12 bits of zero in the low order position of a central memory word that are used to terminate a line of coded information. A record with such a terminator in CYBER Record Manager is a zero-byte record (Z type record).

Files created interactively at a terminal and by commands that manipulate coded lines contain zero-byte terminated records. The image of cards input through a card reader also has such a terminator.

In 6-bit display code, two colons create 12 bits of zeros in the 64-character set. If two consecutive colons occur in a file that contains zero-byte terminated records, they might be stored in the lower order portion of a word and create a zero-byte terminator.

CIO-coded I/O operations, system commands, and dayfile processing macros use files whose lines are terminated by a zero byte.

Zero-Length PRU

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

INTERPRETIVE MODE READING AND WRITING OF EXTENDED MEMORY

D

Interpretive mode processing of extended memory read and write operations gives the COMPASS programmer an effective means of breaking up large block extended memory transfers and processing recoverable extended memory errors. The efficiency of long extended memory transfers tends to be degraded because of PP-initiated exchange jumps which force extended memory transfers to be completely restarted. Interpretive mode processing breaks up large blocks into smaller, 400_8-word blocks, thereby minimizing the effects of these exchanges. Extended memory transfer errors are retried as a block transfer and then as single word transfers if necessary. If the error is recovered, it is logged in the system error log and is transparent to the user program. Unrecoverable errors are also logged and must be processed by the user program.

The interpretive routines are available on common deck COMCECS for absolute COMPASS programs and as relocatable routines in SYSLIB for relocatable COMPASS programs. Additionally, common deck COMCECM is provided to redefine the RE and WE COMPASS instructions. COMCECM is also available on systems text ECSTEXT. Thus, for absolute COMPASS programs, the user must either make specific calls to both common decks or call COMCECS and specify S=ECSTEXT on the COMPASS command (refer to the COMPASS Reference Manual). For relocatable COMPASS programs, you need only specify ECSTEXT as an alternate systems text on the COMPASS command.

Programs using interpretive mode reading and writing of extended memory do so with the usual RE and WE COMPASS instructions. If, while in interpretive mode, you desire to perform noninterpretive reading and writing of extended memory, the RD and WT instructions must be used. These instructions are defined on common deck COMCECM. These instructions read and write extended memory directly while in interpretive mode, as in the normal execution of the RE and WE instructions.

The instructions defined in ECSTEXT and common deck COMCECM are in the following formats.

```
inst    Bj

inst    K

inst    Bj+K
```

Instruction inst is one of the following.

inst	Description
RE	Read extended memory in interpretive mode
WE	Write extended memory in interpretive mode
RD	Read extended memory noninterpretively in interpretive mode
WT	Write extended memory noninterpretively in interpretive mode

SPECIAL USER INFORMATION E

This appendix provides special information available to the applications programmer. The following topics are described.

- Job communication area
- Exchange package area
- File types and job origin codes
- Device types
- EJT status and connection status codes
- Service class codes
- EJT job scheduling data formats
- Error flags

JOB COMMUNICATION AREA

Figure E-1 illustrates the first 100_8 words of the job's field length.

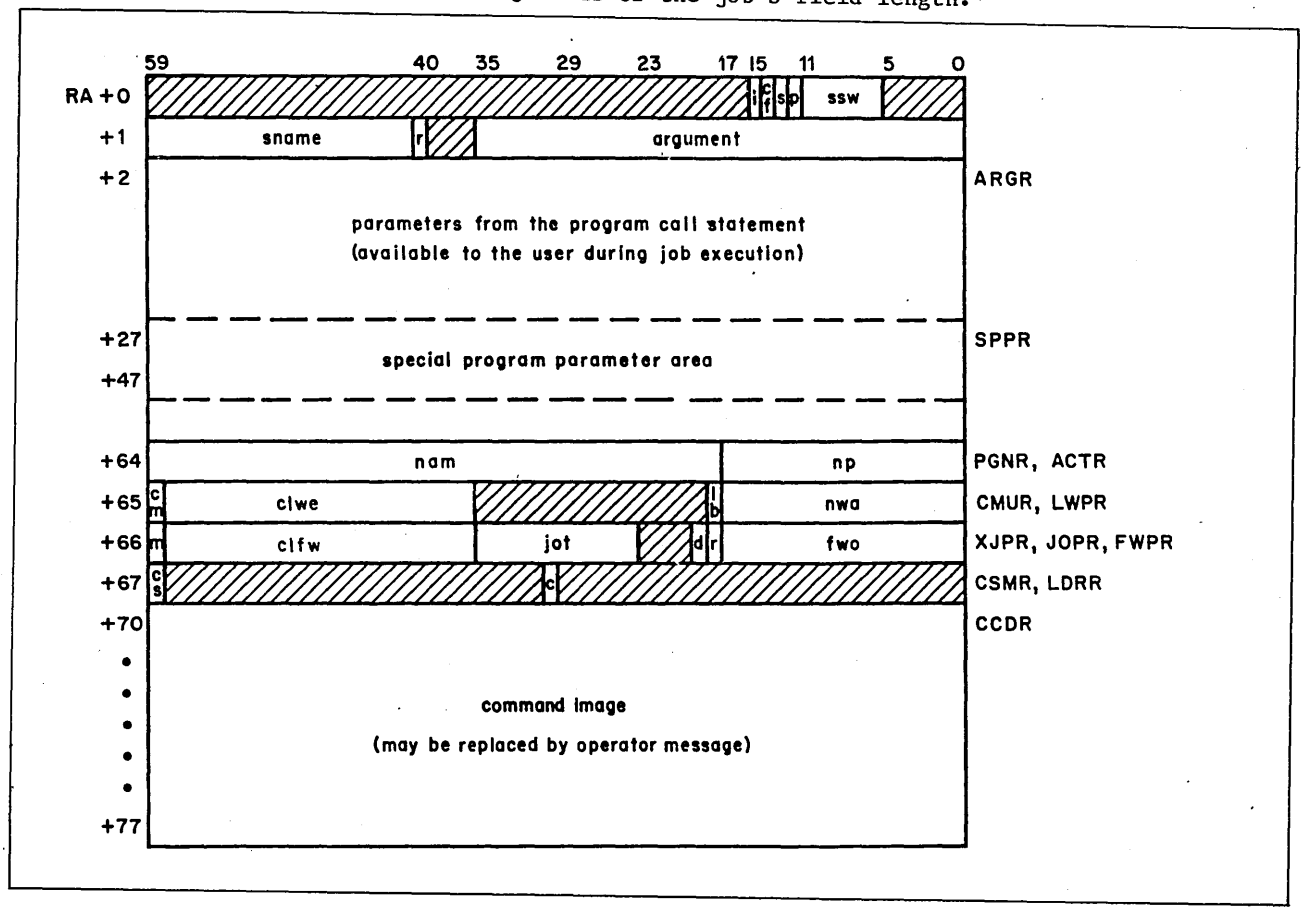


Figure E-1. Job Communication Area

Word	System Identifier†	Bit(s)	Field	Significance
RA+0		59-16	reserved	Reserved.
		15	i	Subsystem idledown flag.
		14	cf	CFO bit.
		13	s	Status bit for pause flag.
		12	p	Pause flag.
		11-06	ssw	Sense switches.
		05-00	reserved	Reserved.
RA+1		59-41	sname	System request name (such as CIO).
		40	r	Auto recall flag.
		39-36	unused	Reserved for future system use.
		35-00	arguments	Parameters passed to that portion of the system that processes the sname request.
RA+2 through RA+63$_8$	ARGR	59-00	params	Parameters from the program call command; available to the user during execution.
RA+64$_8$	PGNR	59-18	nam	Name of program called by command.
	ACTR	17-00	np	Number of parameters in command call.
RA+65$_8$	CMUR	59	cm	Set if the compare/move unit (CMU) is present.
	LWPR	58-36	clwe	LWA+1 of loadable area in extended memory.
		35-19	unused	Reserved for future system use.
		18	lb	Library flag: 0 Load from local file or user library. 1 Load from system library or global library set. (You can distinguish a global library load from a system library load by checking bit 48 of the word returned by the GETLC macro.)

†These symbols are available to a user program if the program calls the SYSCOM macro. The SYSCOM macro is available in both SYSTEXT and NOSTEXT.

Word	System Identifier†	Bit(s)	Field	Significance
		17-00	nwa	Address of next word available for loading. If Common Memory Manager is loaded in the job's field length, bits 17 through 0 contain the complement of the address of the next word available for loading (refer to the Common Memory Manager Reference Manual).
RA+66$_8$	XJPR	59	m	Always set to 1 to indicate that the hardware feature CEJ/MEJ is available.
		58-36	clfw	FWA of loadable area in extended memory.
	JOPR	35-24	jot	Job origin type.
		23-20	unused	Reserved for future system use.
		19	d	DIS flag.
		18	r	RSS flag.
	FWPR	17-00	fwo	First word of object program.
RA+67$_8$	CSMR	59	cs	Set if system is running in 64-character set mode.
	LDRR	58-30	unused	Reserved for future system use.
		29	c	Completion flag: 0 Load not completed. 1 Load completed.
		28-00	unused	Reserved for future system use.
RA+70$_8$ through RA+77$_8$	CCDR	59-00	command image	Image of command currently being executed.

† These symbols are available to a user program if the program calls the SYSCOM macro. The SYSCOM macro is available in both SYSTEXT and NOSTEXT.

EXCHANGE PACKAGE AREA

Figure E-2 illustrates the exchange package area for CYBER 180-class models and models 865 and 875. This illustration and the description that follows it are intended for use by the applications programmer and, therefore, do not include descriptions of bits or fields not used by applications programs.

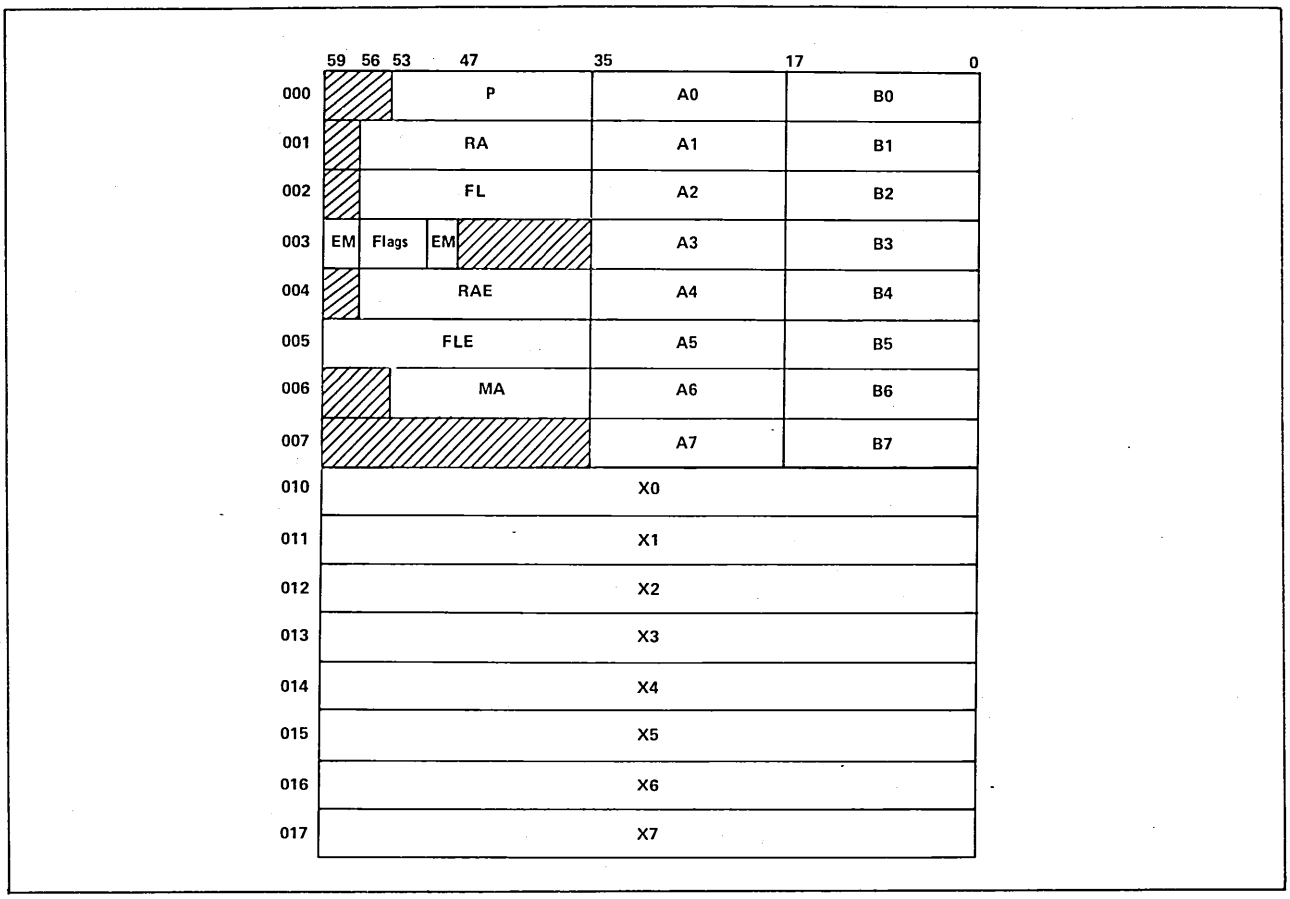


Figure E-2. Exchange Package Area

The exchange package area fields are defined as follows (fields are applicable to all NOS computer systems unless otherwise noted).

Field	Description
P	Program address ($0 \leq P \leq FL-1$).
Ai	Address registers.
Bi	Increment registers.
RA	Reference address for central memory.
FL	Field length for central memory.

Field	Description
EM	Exit modes. An exit mode is selected or disabled by setting or clearing the appropriate bit (refer to the MODE macro).

Bit(s)	Meaning When Bit is Set
50	Exit on indefinite operand.
49	All models except model 176: exit on operand out of range (infinite operand).
48	All models except model 176: exit on address out of range. Model 176: exit on underflow.†

Flags

Bit	Description
56	UEM enabled.
55	Expanded addressing select (ESM mode flag).
54	Extended block copy flag.
53	Reserved.
52	Stack purging on models 815, 825, 835, 845, and 855 (reserved on other models).
51	Reserved.

COND — Error condition flags (model 176 only).

Bit(s)	Description
47, 46	Reserved for Control Data.
45	Extended memory block range condition.
44	CM block range condition.
43	Extended memory direct range condition.
42	CM direct range condition.
41	Program range condition.
40	Not used.
39	Step condition.
38	Indefinite condition.
37	Overflow condition.
36	Underflow condition.

RAE — Reference address for extended memory.

FLE — Field length for extended memory. Bit 56 is zero for all applications programs.

Xi — Operand registers.

†An address out-of-range condition always causes a program exit on a model 176.

EJT CONNECTION STATUS CODES

The following EJT connection status codes are valid. (These mnemonics are defined in the system OPL common deck COMSEJT.)

Code	Value (Octal)	Description
NICS	0	Not interactive. There is no relationship between a job with this connection status and IAF. All batch, remote batch, and system origin jobs are assigned this connection status. A job in this state is not recoverable.
DTCS	1	Detached. A logical connection existed between a job with this type of connection status and an interactive device. However, the logical connection no longer exists (due to a detach command, line disconnect, IAF abort, or other condition). A job in this state continues to execute until an interactive request is issued. A job in this state is recoverable.
OLCS	2	On-line. A logical connection exists between a job with this type of connection status and an interactive device.
--	3-13	Reserved for Control Data.
--	14-17	Reserved for installations.

EJT JOB STATUS CODES

The following EJT job status codes are valid. (These mnemonics are defined in the system OPL common deck COMSEJT.)

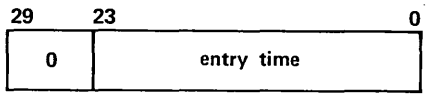
Code	Value (Octal)	Description
PRJS	0	Preinitial job step. A job is in this state after it has been assigned to a control point for the first time. A job in this state is available to be scheduled to a control point.
EXJS	1	Executing. The job is currently at a control point.
ROJS	2	Scheduler rollout. The job was rolled out by the job scheduler because the job's scheduling priority is lower than the priority of another job in the rollout queue. A job in this status is waiting to be rolled in to a control point.
SIJS	3	SCP rollin. An SCP swapin request was issued to roll in the job. The job is currently waiting to be rolled in to a control point.
SOJS	4	SCP rollout. The job was rolled out due to an SCP swapout request. The job remains rolled out until its job status is changed to SIJS.

Code	Value (Octal)	Description
TOJS	5	Timed/event rollout. The job was rolled out because a timed/event was posted. The job remains rolled out until its job status is changed to ROJS.
IOJS	6	Interactive rollout. The job rolled out due to interactive I/O processing. The job remains rolled out until its job status is changed to ROJS.
DOJS	7	Disabled rollout. The job was rolled out because it was a TXOT job with no further commands to process. The job remains rolled out until the job status is changed to ROJS.
SUJS	10	Suspended rollout. The job was rolled out because it was suspended. The job remains rolled out until the job status is changed to ROJS.
ERJS	11	I/O error on rollout. The system encountered a mass storage read error while attempting to roll in a job. The job is left in this job state until a level 0 deadstart.
NVJS	12	Reserved.
--	13-30	Reserved for Control Data.
--	31-37	Reserved for installations.

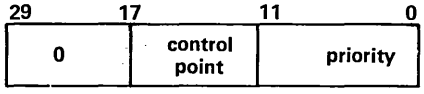
EJT Scheduling Data Field

The EJT scheduling data field is returned for executing jobs in a QAC PEEK request. The information in this field depends on the current stage of execution of the job.

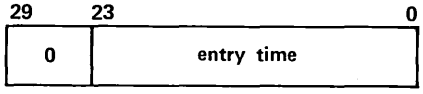
PRJS (Preinitial Job Step)



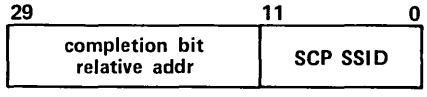
EXJS (Executing)



ROJS (Scheduler Rollout)



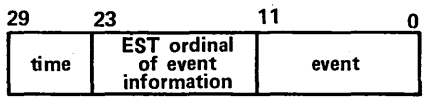
SIJS (SCP Rollin)



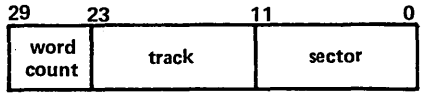
SOJS (SCP Rollout)

Scheduling data field not used.

TOJS (Time/Event Rollout)



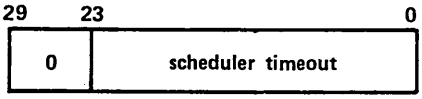
IOJS (Interactive Rollout)



DOJS (Disabled)

Scheduling data field not used.

SUJS (Suspended)



FILE TYPES AND ORIGIN CODES

The following file types and origin codes are used in many NOS system routines.

The following file types are valid. (These mnemonics are defined in NOSTEXT.)

Type	Value (Octal)	Description
--	0-6	Reserved.
ROFT	7	Rollout.
LIFT	10	Library.
PTFT	11	Primary terminal.
PMFT	12	Direct access permanent file.
FAFT	13	Fast attach file.
SYFT	14	System.
LOFT	15	Local.
--	16	Reserved.
INFT	17	Input.
QFFT	20	Deferred routed queue file.
--	21-77	Reserved.

The following job origin types are valid. (These mnemonics are defined in NOSTEXT.)

Type	Value (Octal)	Description
SYOT	0	System.
BCOT	1	Local batch.
RBOT	2	Remote batch (the symbol EIOT is also available but will be deleted in a future NOS release).
IAOT	3	Interactive (the symbol TXOT is also available but will be deleted in a future NOS release).

SERVICE CLASS TYPES

The following service class types are valid. The service class code is the first two letters of each service class type. These mnemonics are defined in the system OPL common deck COMSSCO.

Type	Value (Octal)	Description
SYSC	1	System service class. All jobs initiated by the system, except for maintenance jobs and subsystems, are assigned to this service class.
BCSC	2	Batch service class.
RBSC	3	Remote batch service class.
TSSC	4	Interactive service class.
DISC	5	Detached interactive service class.
NSSC	6	Network supervisor service class.
SSSC	7	Subsystem service class.
MASC	10	Maintenance service class.
CTSC	11	Communication task service class. All SCOPE 2 Station Facility spun-off task (SPOT) jobs and RHF service jobs are assigned to this service class.
IOSC	12	Installation service class 0.
I1SC	13	Installation service class 1.
I2SC	14	Installation service class 2.
I3SC	15	Installation service class 3.
DSSC	77	Deadstart sequencing service class.

Default Service Classes

The system assigns an initial default service class to each job and each output file. The default value depends on the job or file's origin type. Your site may choose to change its default service classes from these released defaults.

Origin	Default Service Class
SYOT	SYSC
BCOT	BCSC
RBOT	RBSC
IAOT	TSSC

DEVICE TYPES

The following device types are supported by NOS:

Mnemonic	Code (Octal)	Display Equipment
DB	0402	885-42 Disk Storage Subsystem (one to three units, full-track).
DD	0404	834 Disk Storage Subsystem (one to eight units, full-track).
DE	0405	Extended memory.
DIn	0411	844-21 Disk Storage Subsystem (one to eight units, half-track).
DJn	0412	844-4x Disk Storage Subsystem (one to eight units, half-track).
DKn	0413	844-21 Disk Storage Subsystem (one to eight units, full-track).
DLn	0414	844-4x Disk Storage Subsystem (one to eight units, full-track).
DMn	0415	885-1x Disk Storage Subsystem (one to three units, half-track).
DP	0420	Distributive data path to extended memory.
DQn	0421	885-1x Disk Storage Subsystem (one to three units, full-track).
DV	0426	819 Disk Storage Subsystem (one unit, full-track; model 176 only).
DW	0427	819 Double Density Disk Storage Subsystem (one unit, full-track; model 176 only).
MT	1524	Seven-track magnetic tape drive.
NE	1605	Null equipment.
NT	1624	Nine-track magnetic tape drive.
TT	2424	Interactive terminal.

ERROR FLAGS

NOS sets the following error flags in the user job's control point area. The user program can refer to these flags during error exit processing (refer to the EREXIT macro in section 6).

Error Flag (Octal)	Mnemonic†	Description
1	TIET	User break 1.
2	TAET	User break 2.
3	ARET	Arithmetic error.
4	ITET	SCP invalid transfer address.
5	PSET	A program stop was encountered by the CPU.
6	PPET	PP abort. A PP program requested that the job be aborted (CIO or PFM, for example).
7	CPET	CPU abort. The job issued an ABT request.
10	PCET	PP call error. The job called a nonexistent or illegal system request.
11	--	Reserved for installations.
12	MLET	Message limit.
13	TLET	Time limit. Job is allowed an additional 1 to 10 seconds of CPU time for error processing.
14	FLET	File limit. The job attempted to assign more active files to the job than are allowed by the validation parameter.
15	TKET	Track limit. The job requested mass storage space on a device with none available.
16	SRET	SRU limit. The job is allowed an additional 10 SRUs to complete error exit processing.
17	FSET	Forced error.
20	RCET	Job hung in auto recall.
21	ODET	The operator dropped the job.
22	IDET	Idle down.
23††	--	Reserved for installations.
24	RRET	Job rerun.
25	DRET	Deadstart rerun.
26	STET	Suspension timeout (EJT error flag only).
27	OKET	Operator killed job.
30	—	Reserved.
31	SVET	Security conflict.
32	SSET	Subsystem abort.
33	ECET	ECS parity error.
34	PEET	CPU or CM parity error.†††

†In order to use the mnemonic in a COMPASS program, the program must contain an SST pseudo-instruction, and you must specify either systems text PPTEXT or NOSTEXT.

††The symbol SPET has a value of 23_8. Error flags whose value is greater than that of SPET are considered special errors.

†††Applicable to CYBER 170 Computer Systems only.

Error Flag (Octal)	Mnemonic†	Description
35	SYET	System abort.
36	RAET	Recovery abort.
37	RSET	Subsystem recovered during level 3 deadstart.
40	ORET	Override of error condition.
41-77	--	Reserved.

Errors greater than or equal to SPET (23_8) are considered special by certain system routines. Examples of this are as follows.

- The job advancement routine 1AJ does not invoke buffer flushing when an error occurs that is SPET or greater.
- Read and write tape error recovery is terminated by an error greater than or equal to SPET.
- CIO terminates a skip operation prior to completion if the error is SPET or greater.
- Waiting for a tape to be mounted and waiting for track limit terminate only on operator drop (ODET) or an error of value SPET or greater.

†In order to use the mnemonic in a COMPASS program the program must contain an SST pseudo-instruction, and you must specify either systems text PPTEXT or NOSTEXT.

COMMENT SHEET

MANUAL TITLE: CDC NOS Version 2 Reference Set, Volume 4 Program Interface

PUBLICATION NO.: 60459690 **REVISION:** G

NAME: _____

COMPANY: _____

STREET ADDRESS: _____

CITY: _____ **STATE:** _____ **ZIP CODE:** _____

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

☐ Please Reply ☐ No Reply Necessary

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

CUT ALONG LINE

AA3419 REV. 4/79 PRINTED IN U.S.A.

OLD 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

Technology & Publications Division
ARH219
4201 North Lexington Avenue
St. Paul, Minnesota 55126-6198

CUT ALONG LINE

OLD FOLD