SYSTEM SESSION ONLY

S: DCU ␣ <USERID>,<DISK NUMBER> declare new userid

S: DLU ␣ <USERID>,<DISK NUMBER> delete old userid
not for 1st user on /FO

S: LIC ␣ <DISK NUMBER> catalog is listed on TTY.

S: PRC ␣ <DISK NUMBER> catalog is printed on lineprinter

S: RSU ␣ <DISK NUMBER> put content of /L file onto the disk
in order to replace the supervisor.

S: SVD ␣ <DISK NUMBER 1>,<DISK NUMBER 2> copy disk 1 to 2.

- SVD /FX, /FO not allowed.
- label of disk 2 is not destroyed

NORMAL SE

BEGIN OF SESSION

DATE: DD␣MM␣YY (CR)(LF)

TIME: HH␣MN␣SS (CR)(LF)

(LF)(CR) clock is not started.

USERID: <DISK NUMBER>,<USERID>

<USERID> system scans all disks catalogs starting with /FO unit.

SYSTEM 1st user of disk/FO will be used. means that we are in system session.

INPUT COMMAND I/O ERROR : error during read userid operation.

action: enter a new userid on TTY

I/O ERROR error during loading of disk allocation table from disk to core

action: enter userid again from TTY.

USERID UNKNOWN this userid has not been found on the disk(s)

action: enter userid again from TTY.

NORMAL SESSION

S: RDS␣[<FILE CODE>] copy source or seq. data file from source input unit or from seq. input unit <file code> to disk as a /S file.

S: RDO␣[FILE CODE>] copy object from { obj. input unit / seq " " <file code> } to disk as a /O file.

S: RDA␣<Disk FILE CODE>[;<INPUT FILE CODE>]

read data from /E1 (no parameter) or INPUT file code to a temporary user file.

S: PRT␣L<FILE CODE>|/S|/S,<NAME>|<NAME>]

[,<line nb 1>],[<line nb 2>] print disk file on the print unit.

disk file: . source file /S,<Name>
. user data file <Name>
. temp. user data file <file code>
. temp source file: /S

S: LST␣L<file CODE>|/S|/S,<NAME>|<NAME>]

[,<line nb 1>],[<line nb 2>] print disk file on the TTY

disk file: same as for PRT

S: PRD␣[/ØB] directory of user library to print unit (no /ØB)
names of obj modules " " " (/ØB)

$: LSD ␣ [/ØB] no /ØB : directory of user library to T. y
/ØB : names of obj. modules " "

$: PCH ␣ [/<FILE CODE> | /S | <NAME> | <NAME>, /S]
punch a disk file on the punch unit.
disk file : - <NAME>, /S source file
userdata file : <NAME>
temp. user data file <FILE CODE>
temp source file : /S

$: POB ␣ [<NAME>] punch an obj. module from disk. <Name> param.
punch whole temp file /O

$: PLD ␣ <NAME> [,/L] punch a load file <NAME> param
from temp. file /L<NAME>

$: DMF ␣ [/<FILE CODE> | /O | /L | <NAME>] [,<SECT NB 1>],
[,<SECT NB 2>] hexadec. dump on print unit.
(SYST SYST parameter)

<NAME> PARAMETER : file from hbr. ON-LINE USER
<FILE CODE>
/O
/L } SYSTEM DEBUGG.

$: INC ␣ <NAME> [,<USERID>] [/FX]
COPY OF OBJ MODULE TO TEMPORARY FILE /O

<NAME> FROM LIBRARY OF ON-LINE USER
/OBJECT WHOLE LIBR. OF OBJ.
<USERID> [,/FX]

S: MOV␣<NAME>,L/S|/L|<FILE CODE>]⌈,<USERID>⌉⌈,/FR.

MOVE A FILE FROM USER LIBR <USERID>⌈,FX⌉ param TO A

TEMPORARY FILE /S
/L
<FILE CODE> USER DATA FILE

S: SCR␣⌈&/S|/O|/L|<FILE CODE>⌉ ALL USER ASSIGNMENTS ARE Removed, so SYSTEM is reset at begin of the session.

EXCEPT FOR FILE CODES 01→09 /E0→/FF

S: KPF␣⌊/S|/O|/L|<FILE CODE>⌋⌈,<NAME>⌉

Keep a file or obj. module that was built as temporary.

KPF␣/S Keep a file with a SOURCE PROGRAM. (cfr. IDENT NAME)

KPF␣/S,<NAME> keep a file with a file name = <NAME>

KPF␣/L,<NAME> " " "

KPF␣<FILE CODE>,<NAME>

} OLD VERSION WITH SAME NAME IS DELETED.

KPF␣/O all obj. ~~files~~ modules of /O file are kept on user obj file

KPF␣/O,<NAME> only the module with that name is kept on user obj. file. (old versions with same name are deleted.).

S: DEL ␣ [<NAME> | /OB] [,[/S|/O|/L]]

delete from the library a file or an obj. module.

S: ASG ␣ <FILE CODE>, [<FILE CODE> | <DEVICE NAME>]

[,<NAME>] [,<USERID>] [,/FX]

ASSIGN A FILE CODE TO A PERIPH. UNIT <DEVICE NAME>

DISK FILE

TEMPOR. AREA

ASG ␣ <FILE CODE>, <FILE CODE> equate the assignments for two specified file codes

ASG ␣ <FILE CODE>, <DEVICE NAME> Assign a file code to a periph.
TP DA

ASG ␣ <FILE CODE>, DK assign a file code " to a disk. For tempory storage.

ASG ␣ <FILE CODE>, DK, <NAME> file code assigned to a library file on disk.

ASG ␣ <FILE CODE>, DK, <NAME>, USERID, /FX assign a file Code to a file of another user library ON pack/FX but this file will be write protected.

S: SEG ␣ <NAME LIST> <NAME LIST>: : <NAME> | <NAME>, <NAMELIST>

$: RUN␣ [<NAME>] [,O] start a program after load from

{ library <Name> param
{ /L

WITH O : run in master mode.

$: BYE Leaving session + re-initialization of system + ask USERID: to start a new session.

$: ASM␣ L /S |<NAME>] [,NL] assemble a source program

{ from library :<NAME> } + listing on print unit
{ from temp. file /S . }

with NL: no listing is required on print unit.

object → /O file.

$: LKE␣ [N|S|U] [,M] [,L DE|DS] [,/<ADDRESS>]

[<start address>]

LKE N: No scan of libraries
S: standard library only
U: user " "

Default: first user libr. then standard.

M: map requested.
DE: keep entry points for debug.
DS: keep only pointers to symbol tables for debug
Default: no symbolic debug option.

/<address>: displacement of blank common from begin of the load module.

default: blank common at end of load module.

<start address> name of start address = entry of one of the modules in the /O file.

default: last start address in /O file encountered.

S: UPD␣<NAME>[,<OUTP. FILE CODE>[/S][,<Inp command file>]

update a source file in library.

<NAME> of source file

UPD␣<NAME>

UPD␣<NAME>,<OUTPUT FILE CODE> file written as UF file

UPD␣<NAME>, /S file written as a /S file

!! DL␣ line 1 [, line 2] delete line(s).

!! IL [line 1] insert line(s)

!! EN terminate an update : /S is created.

!! AB abort current update /S deleted.

S: DEB␣ [<NAME>] call for debug. of load module <NAME>
DEB " " " " /L

S: EDT␣ <NAME>

S: SVU␣ <USERID>, <DISK NUMBER> copy all files of USERID on disk specified (in USER lib of CURRENT session.