

```

T::ECS      MICRO  1,,/=XEC:/
T::DISK     MICRO  1,,/=XDF:/
T::DIR      MICRO  1,,/=XDR:/
T::LINC     MICRO  1,,/=XLC:/
T::TECH     MICRO  1,,/=XTE:/

```

```

*
*
*  MACRO PACKAGE GLOBAL-PARAMETER-SETTING MACRO
*

```

```

MACSET      MACRO      OPERCL, IPLIST, FILETYPE
              IFC       NE, /OPERCL//
OP:::TCL    SET       OPERCL
              ENDIF
              IFC       NE, /IPLIST//
IP:::LIST   SET       IPLIST
              ENDIF
              IFC       EQ, /FILETYPE/ECS/
OP:::TYPE   MICRO      1,,/#T:::FCS#/
              ELSE
OP:::TYPE   MICRO      1,,/#T:::RISK#/
              ENDIF
              ENDM

```

```

*
*
*  CAPABILITY AND DATUM PARAMETER MACROS
*

```

```

CAPAB       MACRO      IPNO, P1, P2
XX          IFC       NE, /P1/*/
              SX6      P1
ZZ          IFC       NE, /P2//  CHECK FOR INDIRECT CAP
              SX7      P2
              LX7      30
              BX6      X6+X7
              MX7      1
              BX6      X6+X7
ZZ          ENDIF
              SA6      IP:::LIST+IPNO
XX          ENDIF
              ENDM

```

```

*
*
BLOCKC      MACRO      IPNO, INDEX, CNT
              SX6      INDEX
              SX7      CNT
              LX7      30
              BX6      X6+X7
              SA6      IP:::LIST+IPNO
              ENDM

```

```

*
*
DATUM       MACRO      IPNO, DAT, XREG
XX          IFC       NE, /DAT/*/
YY          IFC       NE, /XREG//
              BX6      XREG
ZZ          IFC       NE, ///
YY          ENDIF
              SX6      DAT
ZZ          ENDIF
              SA6      IP:::LIST+IPNO
XX          ENDIF
              ENDM

```

```

*
*
BLOCKD      MACRO      IPNO, ADDR, CNT
              SX6      ADDR
              SX7      CNT

```

```

LX7      30
BX6      X6+X7
SA6      IP::LIST+IPNO
ENDM

```

```

#
# SYSTEM CALL MACRO.. SETS UP XJ AND RELATED TRASH
#

```

```

SYSCALL  MACRO      ADDR,INCR,AUTH
+        VFD        12/0,130B,18/ADDR
        IFC        EQ,/AUTH//
-        VFD        12/0,18/INCR
        ELSE
-        VFD        12/1,18/INCR
        JP         *+1
-        VFD        12/0,18/AUTH
        ENDIF
        ENDM

```

```

#
#
# RETURN-PARAMETER AUTHORIZATION
#

```

```

RPAUTH   MACRO      DATA,CAPS
        SPECIFY     DATA
        SPECIFY     CAPS
        ENDM

```

```

#
# CONSTRUCT RETURN AUTH IN TPLIST AT RUN TIME
MKAUTH   MACRO      IPINDEX,DATA,CAPS
        RETSPEC     IP::LIST+IPINDEX,DATA
        RETSPEC     IP::LIST+IPINDEX+1,CAPS
        ENDM

```

```

#
# WHERE, LOCATION, COUNT
RETSPEC  MACRO      WHERE,LOCATION,COUNT
AB       IFC        EQ,/LOCATION//
        SX6         0
BA       IFC        NE,///
AB       ENDIF
        SX6         LOCATION
        SX7         COUNT
        LX7         30
        BX6         X6+X7
BA       ENDIF
        SA6         WHERE
        ENDM

```

```

#
#
# SPECIFY
SPECIFY  MACRO      LOCATION,NUMBER
        IFC        EQ,/LOCATION//
        VFD        60/0
        ELSE
+        VFD        15/0,15/NUMBER,12/0,18/LOCATION
        ENDIF
        ENDM

```

```

#
#
#
#
# OPEN DISK FILE (READ ONLY)
#

```

```

OPENRO   MACRO      FILE,AUTH
        IFC        EQ,/OP::TYPE#/#T::DISK#/
        CAPAB      0,OP:::CL,#T::DISK#OPRO
        CAPAB      1,FILE
        SYSCALL    IP::LIST,0,AUTH
        ENDIF
        ENDM

```

```

*
*
* OPEN DISK FILE (READ/WRITE)
*
OPENRW      MACRO      FILE*AUTH
            IFC        EQ,/*OP::TYPE#/#T::DISK#/
            CAPAB      0*OP:::CL,*T::DISK#OPRW
            CAPAB      1*FILE
            SYSCALL    IP::LIST,0,AUTH
            ENDIF
            ENDM

*
*
* CLOSE DISK FILE
*
CLOSE       MACRO      DFILE
            IFC        EQ,/*OP::TYPE#/#T::DISK#/
            CAPAB      0*OP:::CL,*T::DISK#CLOS
            CAPAB      1*DFILE
            SYSCALL    IP::LIST,0
            ENDIF
            ENDM

*
* PSEUDO-CLOSE FILE
*
PCLOSE      MACRO      EFILE
            CAPAB      0*OP:::CL,*T::DISK#EPCL
            CAPAB      1*EFILE
            SYSCALL    IP::LIST,0
            ENDM

*
*
* CREATE FILE
*
CRFILE      MACRO      ALLOC,CLISTX,NLEVELS,SHAPE
            CAPAB      0*OP:::CL,*OP::TYPE#CFIL
            CAPAB      1*ALLOC
            DATUM      2*CLISTX
            IFC        EQ,/*OP::TYPE#/#T::DISK#/
            BLOCKD     3*SHAPE,NLEVELS
            ELSE
            DATUM      3*NLEVELS
            DATUM      4*SHAPE
            ENDIF
            SYSCALL    IP::LIST,0
            ENDM

*
*
* CREATE FILEBLOCK
*
CRBLK       MACRO      FILE*FIL*AD,WDCOUNT,FRET
            CAPAB      0*OP:::CL,*OP::TYPE#CBLK
            CAPAB      1*FILE
            DATUM      2*FILE*AD
            IFC        EQ,/*OP::TYPE#/#T::DISK#/
            DATUM      3*WDCOUNT
            SYSCALL    IP::LIST,1
            JP         FRET
            ELSE
            SYSCALL    IP::LIST,0
            ENDIF

```

ENDM

*

*

* ATTACH DISK FILEBLOCK

*

```
ATTACH    MACRO      FILE,FILEAD,CNT,FRET
          IFC        EQ,/*OP:::TYPE#/*T::DISK#/
          CAPAB      0,OP:::CL,#T::DISK#ATCH
          CAPAB      1,FILE
          DATUM      2,FILEAD
          DATUM      3,CNT
          SYSCALL    IP::LIST,1
          JP         FRET
          BSS        0
          ENDIF
          ENDM
```

*

*

* DETACH DISK FILEBLOCKS

*

```
DETACH    MACRO      FILE,FILEAD,CNT
          IFC        EQ,/*OP:::TYPE#/*T::DISK#/
          CAPAB      0,OP:::CL,#T::DISK#DTCH
          CAPAB      1,FILE
          DATUM      2,FILEAD
          DATUM      3,CNT
          SYSCALL    IP::LIST,0
          ENDIF
          ENDM
```

*

*

* READ FILE

*

```
READ      MACRO      FILE,FILEAD,CMAD,CNT,FRET
          CAPAB      0,OP:::CL,#OP:::TYPE#READ
          CAPAB      1,FILE
          DATUM      2,FILEAD
          DATUM      3,CMAD
          DATUM      4,CNT
          SYSCALL    IP::LIST,1
          JP         FRET
          BSS        0
          ENDM
```

*

*

* WRITE FILE

*

```
WRITE     MACRO      FILE,FILEAD,CMAD,CNT,FRET
          CAPAB      0,OP:::CL,#OP:::TYPE#WRIT
          CAPAB      1,FILE
          DATUM      2,FILEAD
          DATUM      3,CMAD
          DATUM      4,CNT
          SYSCALL    IP::LIST,1
          JP         FRET
          BSS        0
          ENDM
```

*

*

* TEST ECS FILEBLOCK STATUS (CLEAN/DIRTY)

*

```

TSTBLK MACRO FILE,FILEAD
IFC EQ,/#OP:::TYPE#/#T:::ECS#/
CAPAB 0,OP:::CL,#T:::ECS#TBLK
CAPAB 1,FILE
DATUM 2,FILEAD
SYSCALL IP::LIST,0
ENDIF
ENDM

```

```

*
*
* PROBE FOR FILEBLOCK
*

```

```

PROBE MACRO FILE,FILEAD
CAPAB 0,OP:::CL,#OP:::TYPE#PROB
CAPAB 1,FILE
DATUM 2,FILEAD
SYSCALL IP::LIST,0
ENDM

```

```

*
*
* MOVE FILEBLOCK
*

```

```

MVBLK MACRO SOURCEF,SOURCEA,SINKF,SINKA,FRET
CAPAB 0,OP:::CL,#OP:::TYPE#MBLK
CAPAB 1,SOURCEF
DATUM 2,SOURCEA
CAPAB 3,SINKF
DATUM 4,SINKA
SYSCALL IP::LIST,1
JP FRET
BSS 0
ENDM

```

```

*
*
* DESTROY FILEBLOCK
*

```

```

DSBLK MACRO FILE,FILEAD,P3,P4
CAPAB 0,OP:::CL,#OP:::TYPE#DBLK
CAPAB 1,FILE
DATUM 2,FILEAD
IFC EQ,/#OP:::TYPE#/#T:::DISK/
DATUM 3,P3 FOR DISK FILES, P3 = WD CNT
SYSCALL IP::LIST,1
JP P4 FOR DISK FILES, P4 = FRET LABEL
ELSE
SYSCALL IP::LIST,1
JP P3 FOR ECS FILES, P3 = FRET LABEL
ENDIF
BSS 0
ENDM

```

```

*
*
* DESTROY FILE
*

```

```

DSFILE MACRO FILE
CAPAB 0,OP:::CL,#OP:::TYPE#DFIL
CAPAB 1,FILE
SYSCALL IP::LIST,0
ENDM

```

```

*
*
* CREATE ALLOCATION BLOCK
*

```

```

CRALBLK MACRO FATHER,C,ISTX
CAPAB 0,OP:::CL,#T:::ECS#CRAB
CAPAB 1,FATHER

```

```
DATUM      2:CLISTX
SYSCALL    IP::LIST.0
ENDM
```

```
*
*
* DISPLAY ALLOCATION BLOCK
```

```
DISPABLK  MACRO      ALLOCBLK,BUFFER,BUFSIZE
CAPAB      0:OP:::ICL,*T::ECS#DSAB
CAPAB      1:ALLOCBLK
DATUM      2:BUFFER
DATUM      3:BUFSIZE
SYSCALL    IP::LIST.0
ENDM
```

```
*
*
* MOVE SPACE BETWEEN ALLOCATION BLOCKS
```

```
MVSPACE   MACRO      SOURCE,SINK,AMOUNT
CAPAB      0:OP:::ICL,*T::ECS#MVSP
CAPAB      1:SOURCE
CAPAB      2:SINK
DATUM      3:AMOUNT
SYSCALL    IP::LIST.0
ENDM
```

```
*
*
* CREATE CLASS CODE
```

```
CRCLCD    MACRO      CLISTX
CAPAB      0:OP:::ICL,*T::ECS#CCLS
DATUM      1:CLISTX
SYSCALL    IP::LIST.0
ENDM
```

```
*
*
* SET TEMPORARY PART OF CLASS CODE
```

```
SETEMP     MACRO      CLCD,CLISTX,TEMP
CAPAB      0:OP:::ICL,*T::ECS#STMP
CAPAB      1:CLCD
DATUM      2:CLISTX
DATUM      3:TEMP
SYSCALL    IP::LIST.0
ENDM
```

```
*
*
* DISPLAY CLOCKS
```

```
DSPCLK     MACRO      BUFFER
CAPAB      0:OP:::ICL,*T::ECS#CLK
DATUM      1:BUFFER
SYSCALL    IP::LIST.0
ENDM
```

```
*
*
* DISPLAY MAP ENTRY
```

```
DISPMAP    MACRO      SUBP,ENTRY,BUFFER
CAPAB      0:OP:::ICL,*T::ECS#DSAM
CAPAB      1:SUBP
DATUM      2:ENTRY
DATUM      3:BUFFER
SYSCALL    IP::LIST.0
ENDM
```

```

*
* ZERO MAP ENTRY
*
ZMAP      MACRO      SUBP,ENTRY,FILE
          CAPAB      0,OP:::CL,#OP:::TYPE#ZRM
          CAPAB      1,SUBP
          DATUM      2,ENTRY
          CAPAB      3,FILE
          SYSCALL    IP::LIST,0
          ENDM

*
*
* MAKE READ/WRITE MAP ENTRY
*
INMAPRW   MACRO      SUBP,ENTRY,FILE,FILEAD,CMAD,CNT,FRET
          CAPAB      0,OP:::CL,#OP:::TYPE#MMRW
          CAPAB      1,SUBP
          DATUM      2,ENTRY
          CAPAB      3,FILE
          DATUM      4,FILEAD
          DATUM      5,CMAD
          DATUM      6,CNT
          IFC        EQ, FRET//
          SYSCALL    IP::LIST,0
          ELSE
          SYSCALL    IP::LIST,1
          JP         FRET
          BSS        0
          ENDF
          ENDM

*
*
* MAKE READ-ONLY MAP ENTRY
*
INMAPRO   MACRO      SUBP,ENTRY,FILE,FILEAD,CMAD,CNT,FRET
          CAPAB      0,OP:::CL,#OP:::TYPE#MMRO
          CAPAB      1,SUBP
          DATUM      2,ENTRY
          CAPAB      3,FILE
          DATUM      4,FILEAD
          DATUM      5,CMAD
          DATUM      6,CNT
          IFC        EQ, FRET//
          SYSCALL    IP::LIST,0
          ELSE
          SYSCALL    IP::LIST,1
          JP         FRET
          BSS        0
          ENDF
          ENDM

*
*
* CHANGE READ/WRITE MAP ENTRY (I.E. MAP ENTRY ALREADY EXISTS)
*
CHGMAPRW  MACRO      SUBP,ENTRY,FILE,FILEAD,CMAD,CNT,FILE2,FRET
          CAPAB      0,OP:::CL,#OP:::TYPE#CMRW
          CAPAB      1,SUBP
          DATUM      2,ENTRY
          CAPAB      3,FILE
          DATUM      4,FILEAD
          DATUM      5,CMAD

```



```

DATTM      6*CNT
CAPAB      7*FILE2
IFC        EQ, /FRET//
SYSCALL    IP::LIST.0
ELSE
SYSCALL    IP::LIST.1
JP         FRET
BSS        0
ENDIF
ENDM

```

```

*
*
* CHANGE READ-ONLY MAP ENTRY (I.E. MAP ENTRY ALREADY EXISTS)
*

```

```

CHGMAPRO  MACRO      SUBP,ENTRY,FILE,FILEAD,CMAD,CNT,FILE2,FRET
CAPAB      0*OP:::CL,#OP:::TYPE#CMRO
CAPAB      1*SUBP
DATTM      2*ENTRY
CAPAB      3*FILE
DATTM      4*FILEAD
DATTM      5*CMAD
DATTM      6*CNT
CAPAB      7*FILE2
IFC        EQ, /FRET//
SYSCALL    IP::LIST.0
ELSE
SYSCALL    IP::LIST.1
JP         FRET
BSS        0
ENDIF
ENDM

```

```

*
*
* CREATE EVENT CHANNEL
*

```

```

CREVCH    MACRO      ALLOC,CL,STX,LENGTH
CAPAB      0*OP:::CL,#T:::ECS#CEVC
CAPAB      1*ALLOC
DATTM      2*CLSTX
DATTM      3*LENGTH
SYSCALL    IP::LIST.0
ENDM

```

```

*
*
* SEND EVENT
*

```

```

SENDEV    MACRO      EVCH,EVENT,NOROOM
LOCAL     OUT
CAPAB      0*OP:::CL,#T:::ECS#SEV
CAPAB      1*EVCH
DATTM      2*EVENT
SYSCALL    IP::LIST.0
SX6       X6=3
NG         X6.OUT
IFC        EQ, /NOROOM//
RJ         SYSERR
ELSE
ZR         X6.NOROOM
JP         NOROOM+1
ENDIF
BSS

```



```

*
*
* GET EVENT OR HANG
*
GETEVH    MACRO      EVCH
          CAPAB      0,OP:::CL,#T::ECS#GEVH
          CAPAB      1,EVCH
          SYSCALL    IP::LIST,0
          ENDM

*
*
* GET EVENT OR FRETURN
*
GETEVF    MACRO      EVCH,FRET
          CAPAB      0,OP:::CL,#T::ECS#GEVF
          CAPAB      1,EVCH
          SYSCALL    IP::LIST,1
          JP         FRET
          BSS        0
          ENDM

*
*
* MULTIPLE CAPABILITY PARAMETER MACRO
*
NEXT      MACRO      CNT,PTR,P1,P2,P3,P4,P5,P6,P7,P8
          IFNE      CNT,0
          CAPAB     PTR,P1
          NEXT      CNT-1,PTR+1,(P2),(P3),(P4),(P5),(P6),(P7),(P8)
          ENDIF
          ENDM

*
*
* GET EVENT OR HANG (MULTIPLE EVENT CHANNELS)
*
GETEVMH   MACRO      NEVCHS,EV1,EV2,EV3,EV4,EV5,EV6,EV7,EV8
          CAPAB      0,OP:::CL,#T::ECS#GVMH      NOTE: ↑NEVCHS↑
          DATUM      1,IP::LIST+3                MUST BE ASSEMBLY
          DATUM      2,NEVCHS                     TIME CONSTANT
          NEXT      NEVCHS,3,(EV1),(EV2),(EV3),(EV4),(EV5),(EV6),(EV7),(EV8)
          SYSCALL    IP::LIST,0
          ENDM

*
*
* GET EVENT OR FRETURN (MULTIPLE EVENT CHANNELS)
*
GETEVMF   MACRO      NEVCHS,FRET,EV1,EV2,EV3,EV4,EV5,EV6,EV7,EV8
          CAPAB      0,OP:::CL,#T::ECS#GVMF
          DATUM      1,IP::LIST+3      NOTE: ↑NEVCHS↑ MUST BE
          DATUM      2,NEVCHS          ASSEMBLY TIME CONSTANT
          NEXT      NEVCHS,3,(EV1),(EV2),(EV3),(EV4),(EV5),(EV6),(EV7),(EV8)
          SYSCALL    IP::LIST,1
          JP         FRET
          BSS        0
          ENDM

*
*
* CREATE CAPABILITY LIST
*
CRCLIST   MACRO      ALLOC,CLISTX,LENGTH
          CAPAB      0,OP:::CL,#T::ECS#CCL
          CAPAB      1,ALLOC
          DATUM      2,CLISTX
          DATUM      3,LENGTH
          SYSCALL    IP::LIST,0

```

```

ENDM
*
* COPY CAPABILITY FROM FULL CLIST TO ARBITRARY CLIST
*

```

```

CAPOUT  MACRO      CLIST,CAPX,FULLX
        CAPAB      0*OP:::CL,*T::ECS#COUT
        CAPAB      1*CLIST
        DATUM      2*CAPX
        DATUM      3*FULLX
        SYSCALL    IP::LIST,0
        ENDM

```

```

*
* COPY CAPABILITY FROM ARBITRARY CLIST TO FULL CLIST
*

```

```

CAPIN   MACRO      CLIST,CAPX,FULLX
        CAPAB      0*OP:::CL,*T::ECS#CIN
        CAPAB      1*CLIST
        DATUM      2*CAPX
        DATUM      3*FULLX
        SYSCALL    IP::LIST,0
        ENDM

```

```

*
* COPY CAPABILITY WITHIN FULL CLIST AND DOWNGRADE OPTION BITS
*

```

```

COPYCAP MACRO      SOURCEX,DESTX,OPTMASK
        CAPAB      0*OP:::CL,*T::ECS#MCAP
        CAPAB      1*SOURCEX
        DATUM      2*DESTX
        DATUM      3*OPTMASK
        SYSCALL    IP::LIST,0
        ENDM

```

```

*
* CLEAR (ZERO) A CAPABILITY LOCATION
*

```

```

ZEROCAP MACRO      INDEX
        CAPAB      0*OP:::CL,*T::ECS#ZCAP
        DATUM      1*INDEX
        SYSCALL    IP::LIST,0
        ENDM

```

```

*
* DISPLAY CAPABILITY
*

```

```

DISPCAP MACRO      CAPX
        CAPAB      0*OP:::CL,*T::ECS#DSCP
        CAPAB      1*CAPX
        SYSCALL    IP::LIST,0
        ENDM

```

```

*
* GET CAPABILITY-CREATING AUTHORIZATION FOR NEW TYPE CAPABILITIES
*

```

```

GETCCA  MACRO      CINDX      GET CAPABILITY CREATING AUTHORIZATION
        CAPAB      0*OP:::CL,*T::ECS#MCCA
        DATUM      1*CINDX
        SYSCALL    IP::LIST,0
        ENDM

```

```

*
* CREATE CAPABILITY OF AUTHORIZED TYPE
*

```

```

CRCAP   MACRO      CINDX,CCAUTH,DATA  MAKE A CAPABILITY
        CAPAB      0*OP:::CL,*T::ECS#CCAP

```

```

DATUM      1: CINDEX
CAPAB      2: CCAUTH
DATUM      3: DATA
SYSCALL    IP::LIST.0
ENDM

```

```

*
* CREATE SUBPROCESS CALL/JUMP OPERATION
*

```

```

CREOPER    MACRO      ALLOC,CINDEX,TYPE,SUBP,NUMPAR
CAPAB      0:OP:::CL,#T::ECS#COPR
CAPAB      1:ALLOC
DATUM      2:CINDEX
DATUM      3:TYPE
CAPAB      4:SUBP
DATUM      5:NUMPAR
SYSCALL    IP::LIST.0
ENDM

```

```

*
* ADD ORDER TO EXISTING OPERATION
*

```

```

ADDORD     MACRO      ALLOC,NEWOP,OLDOP,CLASS,NPAR
CAPAB      0:OP:::CL,#T::ECS#ADOR
CAPAB      1:ALLOC
DATUM      2:NEWOP
CAPAB      3:OLDOP
CAPAB      4:CLASS
DATUM      5:NPAR
SYSCALL    IP::LIST.0
ENDM

```

```

*
* SET PARAM SPECIFICATION TO *USER DATUM*
*

```

```

USRDAT     MACRO      OPER,INDEX
CAPAB      0:OP:::CL,#T::ECS#UDAT
CAPAB      1:OPER
DATUM      2:INDEX
SYSCALL    IP::LIST.0
ENDM

```

```

*
* SET PARAM SPECIFICATION TO *FIXED DATUM*
*

```

```

FIXDAT     MACRO      OPER,INDEX,VALUE
CAPAB      0:OP:::CL,#T::ECS#FDAT
CAPAB      1:OPER
DATUM      2:INDEX
DATUM      3:VALUE
SYSCALL    IP::LIST.0
ENDM

```

```

*
* SET PARAM SPECIFICATION TO *USER CAPABILITY*
*

```

```

USRCAP     MACRO      OPER,INDEX,TYPE,OPTIONS
CAPAB      0:OP:::CL,#T::ECS#UCAP
CAPAB      1:OPER
DATUM      2:INDEX
DATUM      3:TYPE
DATUM      4:OPTIONS
SYSCALL    IP::LIST.0
ENDM

```

```

*
*
* SET PARAM SPECIFICATION TO ↑BLOCK DATA↑
*
BLKDAT  MACRO      OPER,INDEX,MAX
        CAPAB      0:OP:::CL,*T::ECS#BDAT
        CAPAB      1:OPER
        DATUM       2:INDEX
        DATUM       3:MAX
        SYSCALL     IP::LIST,0
        ENDM

```

```

*
*
* SET LOCAL ERROR SELECTION MASK
*

```

```

MYESM   MACRO      POINTER
        CAPAB      0:OP:::CL,*T::ECS#ESML
        DATUM       1:POINTER
        SYSCALL     IP::LIST,0
        ENDM

```

```

*
*
* SET P-COUNTER IN STACK ENTRY
*

```

```

SETPCNT MACRO      SUBP,SKIP,PCNT,FRET
        CAPAB      0:OP:::CL,*T::ECS#SETP
        CAPAB      1:SUBP
        DATUM       2:SKIP
        DATUM       3:PCNT
        SYSCALL     IP::LIST,1
        JP          FRET
        BSS         0
        ENDM

```

```

*
*
* DESTROY SUBPROCESS
*

```

```

DELSUBP MACRO      SUBP
        CAPAB      0:OP:::CL,*T::ECS#DSUB
        CAPAB      1:SUBP
        SYSCALL     IP::LIST,0
        ENDM

```

```

*
*
* DISPLAY SUBPROCESS
*

```

```

DISPSUB MACRO      SUBP,BUFFER,BUFLEN
        CAPAB      0:OP:::CL,*T::ECS#DSSB
        CAPAB      1:SUBP
        DATUM       2:BUFFER
        DATUM       3:BUFLEN
        SYSCALL     IP::LIST,0
        ENDM

```

```

*
*
* VIEW STACK ENTRY
*

```

```

VIEW    MACRO      BUFFER,INDEX
        CAPAB      0:OP:::CL,*T::ECS#DSSE
        DATUM       1:BUFFER
        DATUM       2:INDEX

```

```

SYSCALL    IP::LIST.0
ENDM

*
*
* FIND N+TH SON AND RETURN CAPABILITY
*
FINDSON    MACRO      FATHER.N,CLISTX,FRET
CAPAB      0:OP:::CL,*T::ECS#FSON
CAPAB      1:FATHER
DATUM      2:N
DATUM      3:CLISTX
SYSCALL    IP::LIST.1
JP         FRET
ENDM

*
* RETURN ERROR
*
* RETURN AND GENERATE ERROR FOR CALLER
*
RETErr     MACRO      CLASS,NUMBER
CAPAB      0:OP:::CL,*T::ECS#RERR
CAPAB      1:CLASS
CAPAB      2:NUMBER
SYSCALL    IP::LIST.0
ENDM

*
* RETURN WITH PARAMETERS
*
* RETURN WITH PARAMETERS
*
RETPAR     MACRO      DATA,CAPS
CAPAB      0:OP:::CL,*T::ECS#RPAR
CAPAB      1:DATA
CAPAB      2:CAPS
SYSCALL    IP::LIST.0
ENDM

*
*
* TELETYPE INPUT / OUTPUT
*
TELETYP    MACRO      ACTION,BUFFER,RETAUTH
CAPAB      0:OP:::CL,*T::LINC#ASCII
*
* INPUT A LINE: ↑TELETYP  INLINE,(BUFFER,BUFLEN)↑
AA         IFC
          DATUM      1:1
          BLOCKD     2:0,0
          MKAUTH      3:(BUFFER)
          SYSCALL     IP::LIST.0,IP::LIST+3
          IFC         NE,///
SKIP
AA         ENDIF

*
* OUTPUT A LINE: ↑TELETYP  OUTLINE,(BUFFER,BUFLEN)↑
BB         IFC
          EQ,/ACTION/OUTLINE/
          DATUM      1:2
          BLOCKD     2:BUFFER
          SYSCALL     IP::LIST.0
          IFC         NE,///
SKIP
BB         ENDIF

*
* OUTPUT A CHARACTER: ↑TELETYP  OUTCHAR,CHAR↑
CC         IFC
          EQ,/ACTION/OUTCHAR/
          DATUM      1:3
          BLOCKD     2:IP::LIST+3,1
          DATUM      3:BUFFER
          SYSCALL     IP::LIST.0
          IFC         NE,///
SKIP
CC         ENDIF

*
* EDIT LINE: ↑TELETYP  EDIT,(OLDBUF,LEN),(NEWBUF,LEN)↑
DD         IFC
          EQ,/ACTION/EDIT/
          DATUM      1:4

```

```

BLOCKD 2,BUFFER
MKAUTH 3,(RETAUTH)
SYSCALL IP::LIST,0,IP::LIST+3
IFC NE,///
ENDIF

SKIP
DD
*

ERR NO RECOGNIZED ACTION
ENDIF
ENDM

*
*
* RETURN
*
RETURN MACRO
SYSCALL **1,1
VFD 1/1,29/#T::ECS#RTRN,30/OP:::CL
ENDM

*
*
* FAILURE-RETURN
*
FRETURN MACRO
SYSCALL **1,1
VFD 1/1,29/#T::ECS#FRET,30/OP:::CL
ENDM

*
* SAVE REGISTERS
*
SAVE MACRO
WHERE
SYSCALL **1,2
VFD 1/1,11/0.18/#T::ECS#SAVE,12/0.18/OP:::CL
VFD 60/WHERE
ENDM

*
* RESTORE REGISTERS
*
RESTORE MACRO
WHERE
SYSCALL **1,2
VFD 1/1,11/0.18/#T::ECS#RSTR,12/0.18/OP:::CL
VFD 60/WHERE
ENDM

*
*
*
*
* FIELDS IN FOUR WORD SPACE DESCRIPTORS
*
SPACE DESCRIPTORS APPEAR IN:
1) ECS ALLOCATION BLOCKS
2) DISK ACCOUNTING BLOCKS
3) PROCESS SWAPPED ECS ALLOCATION INFO

*
SD:RESV EQU 0 SPACE RESERVED
SD:INUS EQU 0 SPACE IN-USE
*
SD:TIME EQU 1 TIME OF LAST METER UPDATE
SD:RATE EQU 1 RATE TO RUN METERS
*
SD:CMTR EQU 2 CONTINUOUS CHARGE METER
*
SD:DMTR EQU 3 DISCONTINUOUS CHARGE METER

```

23.05.41.

905 LINES PRINTED BY PRINTER DRIVER ON LP 2.

[illegible]