Table of contents


```
    7-    1       SET command
    9-    1       .       System parameters
   11-    1       .       Real device
   15-    1       .       Prompt
   16-    1       .       Syspassword
   17-    1       .       Endstartup
   18-    1       .       Window
   20-    1       .       Printwindow
   21-    1       .       Priority
   22-    1       .       Maxpriority
   23-    1       .       Error
   23-   12       .       Shutdown
   24-    1       .       Log
   26-    1       .       Logoff
   27-    1       .       Subprocess
   28-    1       .       Kmon
   30-    1       .       LD
   34-    1       .       SL
```

```
                                        .TITLE  TSKST1 -- Keyboard SET Command routines
                                        .ENABL  LC
                                        .DSABL  GBL
                                        .CSECT  TSKST1
                                TSKST1:
                                ;
                                ;  TSKST1 is the portion of TSKMON that contains the code
                                ;  to implement the SET command.
                                ;
                                ;  Copyright 1978, 1979, 1980, 1981, 1982, 1983, 1984, 1985, 1986, 1987, 1988, 1989.
                                ;  S&H Computer Systems, Inc.
                                ;  Nashville, Tennessee
                                ;
                                ;  Macro calls
                                ;
                                        .MCALL  .CSISPC,.TTOUTR,.SRESET
                                        .MCALL  .READW,.TTYIN,.TTYOUT,.PURGE
                                        .MCALL  .CSIGEN,.SAVEST,.REOPEN
                                        .MCALL  .GTLIN,.GTIM,.DATE,.SPFUN
                                        .MCALL  .PRINT,.CLOSE,.LOOKUP
                                        .MCALL  .WRITW,.ENTER,.EXIT
                                        .MCALL  .SERR,.HERR,.FPROT,.GVAL,.PVAL,.DSTAT
                                ;
                                ;  Global definitions
                                ;
                                        .GLOBL  TSKST1,CMDHD,CMDOFF,KDOCIN,SKPSPC,UCLCMD
                                        .GLOBL  DORUN,CMDFRM,CMDDSN,STLGCN,DATTIM,PRGALL
                                        .GLOBL  DLCEMT,ALCDEV,CMDSHO,CMDSET,CMDWHO,CMDMEM,CMDUSE
                                ;
                                ;  Global references
                                ;
                                        .GLOBL  LSW11,$PWKEY,SETCL,SETHST,DVFLAG,DX$NST,TM$MRS
                                        .GLOBL  SC$SUC,SC$WRN,SC$ERR,SC$SEV,SC$FTL,SC$UNC,SC$NON
                                        .GLOBL  SETPRC,SETTTY,STVRFY,STNOVR,PO$SYS,RCLREV,PA$FLG,PA$DTS
                                        .GLOBL  PA$GRC,PA$UKC,PA$DSC,PA$BLD,PA$ULN,PA$DWD,PA$HQL,PA$LET
                                        .GLOBL  EM$PTA,EM$PTU,SBPSUF,CHKCLU,EM$CLX,PA$BEL,EM$OPR,PA$NWD
                                        .GLOBL  SCNOPS,TM$XBK,CL$XLN,CKCLUS,ACRSTR,R50KMN,SJEMT,RJEMT
                                        .GLOBL  PEKEMT,PEKADR,PEKSIZ,TM$NNR,CDBUF,CDGET,TM$IN1,TM$IN2
                                        .GLOBL  SYPSWD,EM$SPL,ACRTXT,JPWDEV,JPWTYP,JPWFLG
                                        .GLOBL  EM$WC0,EM$WC1,EM$WC2,EM$WC3,EM$WCM,P2$CGR
                                        .GLOBL  CXTRMN,R$CHN,R$XCHN,CHNSIZ,C.USED,CL$EPN,CL$EPS,CLEOFS
                                        .GLOBL  EDMALD,EM$STL,EM$IST,CLSFEP,EM$IDR,VPRIDF,SETWRD
                                        .GLOBL  CLRPRV,OPTLST,PFSO,PFCO,PVNPW,PO$SPV,PRIVSO,PRVOPT
                                        .GLOBL  TM$PVA,TM$PVC,PRIVAO,EM$CNO,EM$CPO,EM$CAP,RSTPRV
                                        .GLOBL  CHKEQ,CKACOJ,PO$OPR,CKSYPV,P2$TRM,PRIVC2,PO$NAM,EM$NPR
                                        .GLOBL  INSTBL,INGADR,INGEMT,IIBUF,II$NAM,II$FLG,II$$SZ,EM$NAD
                                        .GLOBL  INSTBN,AF$SCA,AF$NOW,AF$MEM,PO$DBG,PRIVCO,PRVLST
                                        .GLOBL  ABRTAD,ABRTCD,CINFLG,$VNOTT,II$PRV,II$NPV
                                        .GLOBL  TM$RD1,TM$RD2,TM$LCL,TM$GBL,SPACE1,RC$OWN,RC$CNT
                                        .GLOBL  RC$EXC,RC$AGE,RC$AEP,RC$USE,RC$FLG,RC$GBL,RC$NAM
                                        .GLOBL  RC$LEN,RC$PVT,RCBBAS,RCBEND,RC$$SZ,SHRRCB,SHRRCN
                                        .GLOBL  LP$SPD,LP$PAR,LP$ODD,LP$7BT,EM$ICL,PROSLT,RC$LCG
                                        .GLOBL  EM$NPD,EM$ILN,EM$CIP,EM$NSF,EM$IUN,EM$CLN
                                        .GLOBL  EM$ILN,EM$ACL,EM$TSL,EM$CLB,EM$NSL,EM$SLT,EM$SLW
                                        .GLOBL  SLKDON,SLKDOF,EM$UIO,TM$PR1,TM$PR2,TM$LPR,TM$HPR
                                        .GLOBL  TM$HPE,TM$CNG,TM$CDS,TM$CEN,TRMHD1,TRMHD2,AT$DEV
                                        .GLOBL  OPRTXT,CLLINE,LCLTXT,REMTXT,TM$AUT,CLFREE,CLUNIT
```

```
                        .GLOBL  TM$CL0, TM$CL1, TM$CL2, TM$CL3, TM$CL4, TM$CL5, TM$CL6
                        .GLOBL  QHDMS1, QHDMS2, DVSHH1, DVSHH2, DVSHH3, SYASHD, DKASHD
                        .GLOBL  TM$NAD, ALCHD1, ALCHD2, TM$NSD, TM$SDN, LNAME
                        .GLOBL  CORUSR, LSW, $CTRLO, SERFLG, IOABFL, $CHACT, $STSNG
                        .GLOBL  LSTHL, LCLUNT, FSTIOL, LSTIOL, CL$LIX, CW$PRO, CONFG2
                        .GLOBL  CL$RQH, CL$WQH, MAXALC, ALCTBL, ALCEND
                        .GLOBL  AD$DVU, AD$JOB, AD$$SZ, UCIDEF, HANCHN
                        .GLOBL  NEDCHR, LOUTIR, LINIR, LINRTS, CLOTIR
                        .GLOBL  CO$DEF, CL$COL, LCDTYP, SOPALC, SOPDAT, SOPTIM
                        .GLOBL  UTRPAD, JSWLOC, ERRLOC, MAXMEM, MAXPRI
                        .GLOBL  USRSTK, $KINIT, CFSTK, MXJMEM, DFJMEM, EM$HNI
                        .GLOBL  SPUBUF, SXBPNT, MXJADR
                        .GLOBL  TMTOTH, TMTOTL, TMUSRH, TMIOWH, LDMNT, EM$CSE
                        .GLOBL  TMSWTH, TMIDLH, TMIOH, TMSWPH, LDCLEN
                        .GLOBL  WILDFL, $NOIN, $NOWTT, $HITTY
                        .GLOBL  TECO, EDIT, KED, K52, $1STLG, $DIBOL
                        .GLOBL  SH$VAL, SH$NAM, SH$$SZ, SH$RTN, SH$FLG
                        .GLOBL  SO$NVL, SO$OCT, SO$NO, HANENT, HANSIZ
                        .GLOBL  H.CSR, H.VEC, DVSTAT, SFID, ACRSPD, HANPAR, LSTSPL
                        .GLOBL  HAZEL, HAZLFL, HAZLNO, $MLOCK, MDT, GETKCH
                        .GLOBL  LINBUF, LINNXT, LSTACT, PRGTOP, PRGSIZ, KMNHI
                        .GLOBL  KMNTOP, KMNPGS, KMNSTK, KMNSTR, CXTPAG, FSTIOL
                        .GLOBL  LINPNT, LINCNT, LACTIV, LRDTIM, CS$RON
                        .GLOBL  LOTBUF, LOTNXT, LOTPNT, $VTESC
                        .GLOBL  LOTSIZ, LOTSPC, LCOL, $SLKED, ESC, VDBFLG
                        .GLOBL  LAFSIZ, LFWLIM, LINCUR, NUMON, ILSW2
                        .GLOBL  $DBKMN
                        .GLOBL  $CARUP, DOASGN, UKMNAM, $UKMON, LSW9
                        .GLOBL  LSUCF, $CCLRN, VLDSYS, EM$NUK, S$QMIO
                        .GLOBL  KL3CLR, $PRGLK, LSW5, PVON, S$SPND, $AUTO
                        .GLOBL  S$TWFN, S$TTFN, S$OTFN, S$IOFN, S$OTLO
                        .GLOBL  LSTDL, FSTDL, $DETCH, UMSYTP, S$TTSC
                        .GLOBL  $DISCN, LPROJ, LPROG, LUNAME, S$RT, S$LOW
                        .GLOBL  LCPUHI, LCPULO, LCONTM, $CTRLS, $SPLJB, TXTCL
                        .GLOBL  STPFLG, TOTON, USPLCH, SPLCHN, S$HICP
                        .GLOBL  S$INWT, S$OTWT, S$TMWT, S$SFWT, S9600
                        .GLOBL  S$MSWT, CFBUF, CFEND, CCLSAV, KMNCHN
                        .GLOBL  MINTIM, LSECPT, MAXSEC, $EMTTR, VCSHNB
                        .GLOBL  OKFILE, OKFEND, $CLTST, UCISPC, MHNSIZ
                        .GLOBL  CASTBR, CASCBR, CASTBW, CASCUP, MHNSMS
                        .GLOBL  CASTRO, CASTWO, CLTOTL, CO$DTR, CLSFSP
                        .GLOBL  CO$CR, CO$FF, CO$FFO, CO$LC, CO$TAB, CO$CTL
                        .GLOBL  CO$LFI, CO$LFO, CO$BNI, CO$BNO, CL$OPT
                        .GLOBL  CL$LEN, CL$SKP, CL$WID, CL$LIN, PHYMEM
                        .GLOBL  LJSW, CTRLTT, NEWJSW, JSTKND, VIMAGE
                        .GLOBL  USTART, GENTOP, BOTDEV, BOTUNI, CSHALC
                        .GLOBL  $CTRLC, LSW2, $INKMN, CHAIN, UFORM
                        .GLOBL  $SGQO, $SGQ3, LITIME
                        .GLOBL  MAXASN, $CFABT, INDSTA, INDERR
                        .GLOBL  RUNDEV, LNBLKS, CXTBAS, CXTWDS, UHIMEM
                        .GLOBL  $DILUP, CSHDEV, CSHDVN, LNSBLK
                        .GLOBL  LSW3, LSW2S, $DUPRN
                        .GLOBL  $FORM, $TAB, LSCCA, $CFSOT, LOFSPC, R50COM
                        .GLOBL  $PAGE, $SCOPE, $ECHO, $LC, $8BIT, CHKALC
                        .GLOBL  UCHAN, $FORMO, $CFALL, $CFDCC, $CFCCL
                        .GLOBL  LNPRIM, LNMAP, CW$50H, CONFIG, $SUCF
                        .GLOBL  $DOOFF, NUCHN, LRBFIL, CFIND, TALEMT
```

```
                        .GLOBL  C.CSW,C.DEVQ,C.SBLK,NLINES,CO$BBT
                        .GLOBL  CD$NAM,CD$DVU,CD$BAS,CD$JOB,CD$$SZ,CD$$UB
                        .GLOBL  LTSCMD,LNSPAC,CFNEST,UCLNAM
                        .GLOBL  $CFOPN,CFSEND,PBFEND,CFSP,$TTGAG
                        .GLOBL  UFPTRP,SDSFCB,SD$DEL,CFLFL4,$UCLCF
                        .GLOBL  SDFLAG,SD$FLK,SD$WFM,SDFORM,$UCLRN
                        .GLOBL  SDBUF1,SDBLK,NSPLDV,LD$RON,$UCLCM,$UCLCL
                        .GLOBL  LDNAME,LDSIZE,LDFLAG,LDBASE,LDPDEV
                        .GLOBL  LSW8,$SGQ1,$SGQ1A,$SGQ1B,$SGQ1C,$SGQ2,$SGIIO,$SGHIO
                        .GLOBL  $DEFER,CFCHAN,SCHAIN,LDDEVX,$SGALL
                        .GLOBL  CFPNT,CFBLK,$QUIET,DIABFL
                        .GLOBL  DIABNO,VT52NO,LA36NO,LA36FL
                        .GLOBL  LSW4,KL4CLR,SDSKIP,SDBU,SD$BAK
                        .GLOBL  $INCOR,$KED,VQUN1B,VINTIO,VQUN1C
                        .GLOBL  SF$BSY,SFFORM,SD$SNG,SFNMBL,NFRESB
                        .GLOBL  SD$HLD,SF$HLD,CURPRM,PRMPNT,SF$1ST
                        .GLOBL  LSTPRM,PRMBUF,PRMEND,CFSPND
                        .GLOBL  SDFHD,SFFLAG,SFQLNK,CFHOLD,LOGDVU,LOGBAS
                        .GLOBL  LCOL,$QTSET,$TECO,CD$TOP,LOGCHK
                        .GLOBL  $WILD,ERRSEV,UERSEV,PASLIN,LOGBAS,LOGDVU
                        .GLOBL  LSTPL,SDCB,SDCBND,VQUAN0,VQUAN3
                        .GLOBL  VQUAN1,VQUN1A,VQUAN2,VHIPCT,VQUAN0,VQUAN3
                        .GLOBL  DCTRD,DCCRD,DCTWR,DCCWR,ASNSRC
                        .GLOBL  VCORTM,NUMDCD,VNUMDC,KMPRMT,MXPRMT
                        .GLOBL  RDB,RDBEND,RT$NAM,RT$$SZ,CLDEVX,SDDVU
                        .GLOBL  SDNAME,SDCBSZ,LSTSL,LSTATE
                        .GLOBL  TK1VAL,CINDAT,SYSDAT,SYTIMH,SYTIML
                        .GLOBL  BASMAP,LOMAP,HIMAP,JCXPGS
                        .GLOBL  SMRSIZ,SRTSIZ,CSHSIZ,TK1SEC
                        .GLOBL  TSXLN,TSXSIT,GRT1,TRGRET,LICTXT,SUPCOD,NAMTOP,SUMS,SUCS
                        .GLOBL  LPRG1,LPRG2,S$QUSR,S$IOWT,S$SFWT
                        .GLOBL  S$SPDB,S$SPCB,SFUSER,SFFILE,VT200,VT2007,VT2008
                        .GLOBL  LCBIT,LA36,LA120,VT52,VT100,DIABLO,QUME
                        .GLOBL  ADM3A,LTRMTP,LA12FL,LA12NO,VT52FL
                        .GLOBL  VT10FL,VT10NO,QUMEFL,QUMENO,ADM3FL
                        .GLOBL  VT20FL,VT20NO
                        .GLOBL  ADM3NO,SYINDX,SYUNIT,NUMDEV,PNAME
                        .GLOBL  OF$DEV,OF$UNT,OF$FIL,OF$FLG,SYNAME
                        .GLOBL  OF$$SZ,OT$RON,RESDEV,$TAPE
                        .GLOBL  KMNBAS,ODTBAS,$CTRLD
                        .GLOBL  LSW6,$SNWTT,PF$SYS,PF$IOW,$DEBUG
                        .GLOBL  RSR,TSR,LMXNUM,LSTMX,MXDTR,ZCLR,MXCSR
                        .GLOBL  $INDDF,$INDRN,IN$ACT,IN$CNT,IN$CMD,INDSAV
                        .GLOBL  $PHONE,INVEC,LMXLN,MXVEC,$INIT,$DEAD,$HARD
                        .GLOBL  ITRMTP,LMXPRM,LSW7,$INDAB,CFSTS,CF$IND,CF$QUT
                        .GLOBL  CFABLV,MONVEC,LBSPRI,MAXPRI,MXJPRI,LPRI,$SYSPS
                        .GLOBL  LOGCHN,LOGFLG,LOGPTR,LOGBUF,LOGBLK,LF$IN,LF$OUT
                        .GLOBL  LF$OPN,LF$WRT,UCLBLK,UCLDAT
                        .GLOBL  CSHHD,FC$CDX,FC$LNK,FD$NAM,UC$NDC,UC$MDC,CVTUC
                        .GLOBL  CMDBUF,PAUMSG,RDCMD,DKSAV,SYSAV,CVTTAB,RUNHD,SEARCH
                        .GLOBL  INVOPT,FKILL,ABRTCF,ACRFN,XAREA,FILNAM,NOPRG,FPRINT
                        .GLOBL  PUSHCF,TRMSTR,FILNAM,R50DIR,R50SY,R50IND,R50SAV
                        .GLOBL  INDACT,R50DUP,R50PIP,R50KED,R50K52,R50KEX,R50TSX,R50UCL
                        .GLOBL  BLKO,RDERM,R50VIR,NOSTRT,RUNEMT,OVRCOR
                        .GLOBL  BADSAV,LDNAM,NOPRG,NOCIN,SIZVAL,ASKLNM,BADCMD,KCSIBF
                        .GLOBL  ASDEX,KCSIMS,ASNOVF,GTRD50,R50BUF,R50LDO,MNTDEV,DMTARG
                        .GLOBL  DEADEV,CHKMNT,CHKMTX,INFOMT,NOFLAG,MTOPHD,INVOPT,ILLCMD
```

```
                .GLOBL  R50LD, INVLDN, R50DSK, ACRFIL, BDFNAM, LOGASN, MNTFUL, R50LD7
                .GLOBL  TBLOVF, SETHD, CSIMS2, CKPRIV, R50NO, AMBOPT, ACRDEC
                .GLOBL  MAXAVL, PRTDEC, DEVUNT, PNAME, HNBUF, CKTERM
                .GLOBL  ACROCT, HANBSY, CSIMS1, MISSEQ, NOIND, POPCF
                .GLOBL  BADPMT, BADPRI, TOTXT, CRLF, HIPRI, STLGHD, LOGCLS, R50LOG
                .GLOBL  BDLGOP, SPLHLA, NOCCL, LDOPHD, PRTFIX, PRTSPC
                .GLOBL  DLTXT, OCTFIX, PRTTTP, NATXT, SPDTX1, NOTXT, YESTXT, NINTXT
                .GLOBL  PRTUNM, SYHD1, SYHD2, PRTLN, SPACE2, DETTXT, SPACE3, RNMS
                .GLOBL  SWPTX, LOCKTX, SPACE5, PRTDC3, KBMSG, DIVIDE, PRTDC2
                .GLOBL  COLOO, CPUAH, CPUAL, PRTTMV, NOFIL, CMDBUF, CALUCL
                .GLOBL  NOUDC, DEVHD1, ASNHD1, ASNHD2, SHMTH1, SHMTH2, PRTTMD
                .GLOBL  CVDVNM, SPACE6, PRTBUF, PRTFNM, NONEMS, NODAT, NOLDMT
                .GLOBL  SUBARG, EDTFIL, RONTXT, NOTAVL, KBTX, MNFLGS, MNBPC
                .GLOBL  DELSPC, MNBASE, MNTOP, MONHD, MONAR1, NOPMGN, PMBUSY, MONAR2
                .GLOBL  NSWPMS, MAXMTX, CURMTX, CHKDLM, SPLHD, AMBOPT, INVOPT
                .GLOBL  DEVIDL, COAL, ALDEX, COAD, SPACTV, SPWFM, DEVIDL, SPSNG
                .GLOBL  COAL, ALDEX, ALDBLK, COAD, SPACTV, SPWFM, DEVIDL
                .GLOBL  SPSNG, SPFUL, SPCF, SPFLK, NOFIL, SPGEMT, NOOPTT
                .GLOBL  BDLIN, MSGBUF, MSGEND, NOTON, GAGMSG
                .GLOBL  LINFRE, DJABMS, DLMSG, INVTIM, DMTALL
                .GLOBL  SHTMSG, AUTHFN, SPLACT, DOSTOP, OFFEMT, KILEMT, UPTMMS
                .GLOBL  TMTOTH, DIVSOR, TMTOTL, PRTPCT, SUM1, SUM2, SUM3, SUM4
                .GLOBL  SUM5, SUM6, SUM7, OTHRON, SPLPND, STPASK, SRTSMS, CHKTTD
                .GLOBL  SIZEMT, ASNOVF, INVLDM, CSIMS4, MNTARG, HUPARG, R50TT
                .GLOBL  KMNNAM, NOKMON, CCLNAM, OTRMNT, CHKDEV, DMTSUB, CMDCCL
                .GLOBL  SHOHD, SUBTXT, MNTTXT, SRTTXT, TOTMMS, UMSSMS, SSRMAP
                .GLOBL  TSXSMS, USRMMS, JCXSMS, DZTXT, OCTPRT
                .GLOBL  PRTR50, PRTDAT, PRTTOD, PRTTIM, INVDEV, ALFN, R50DK
                .GLOBL  DETHD, DETARG, RUNMS, NOFRDL, R50MON, INVDAT, MUL32, COAF
                .GLOBL  AR$PRJ, AR$PRG, AR$CON, AR$CNT, AR$CPH, AR$CPL, AR$UNM
                .GLOBL  AR$DMY, AR$$SZ, ARNRPB, $SLON, $SLTTY, $SLLET
                .GLOBL  PRTWRN, SLMXLN, VSLEDT, $LOFCF, CSHMSG
                .GLOBL  AF$HIE, AF$NOI, $NOINT, AF$PLK, AF$DBG, DS$DIR
                .GLOBL  AF$IOP, $RNIOP, VOFFTM, VONTM, VTMOUT, VTMIN, VTMLOC, VUSPHN
```

```
                                ;
                                ;  Assembly constants
                                ;
000012          LF      =       12              ;LINE FEED
000015          CR      =       15              ;CARRIAGE RETURN
000040          BLANK   =       40              ;ASCII SPACE
000007          BELL    =       07              ;ASCII BELL
000011          TAB     =       11              ;HORIZONTAL TAB
000014          FF      =       14              ;FORM FEED
000054          COMMA   =       54              ;COMMA
000400          BLKWDS  =       256.            ;# OF WORDS IN DISK BLOCK
132500          WLDNAM  =       132500          ;RAD50 /*/ (WILDCARD)
```

```
                        ;-----------------------------------------------------
                        ;  Macro to cause a fatal error message to be printed.
                        ;
                              .MACRO  FERR    MSG
                              MOV     R5,-(SP)
                              MOV     MSG,R5
                              CALL    FPRINT
                              MOV     (SP)+,R5
                              .ENDM   FERR
                        ;
                        ;-----------------------------------------------------
                        ;  Macro to print a fatal error message, clean up
                        ;  and then jump to RDCMD.
                        ;
                              .MACRO  FABORT  MSG
                              MOV     MSG,R5
                              JMP     FKILL
                              .ENDM   FABORT
                        ;
                        ;-----------------------------------------------------------
                        ;  Macro to print a warning message
                        ;
                              .MACRO  FWARN   MSG
                              MOV     R5,-(SP)
                              MOV     MSG,R5
                              CALL    PRTWRN
                              MOV     (SP)+,R5
                              .ENDM   FWARN
                        ;
                        ;-------------------------------------------------------------
                        ;  Macro to start a standard option table.
                        ;  Name = 1 to 4 character table name.
                        ;  NA = Number of arguments per table entry.
                        ;
                              .MACRO  TBLDEF  NAME,NA
                        NARGS =       NA
                              .CSECT  CMDVS1
                        NAME'HD: .WORD  2*NA
                              .ENDM   TBLDEF
                        ;
                        ;-------------------------------------------------------------
                        ;  Macro to enter an option text name and a set of parameters
                        ;  into the currently open table.
                        ;  STRNG = Ascii name
                        ;  A,B,C = Set of option parameters to store in table with name.
                        ;
                              .MACRO  CMDDEF  STRNG,A,B,C
                              .CSECT  NAMES1
                        L     =       .
                              .ASCIZ  /STRNG/
                              .CSECT  CMDVS1
                              .WORD   L               ;POINTER TO NAME STRING
                              .WORD   A
                        .IIF  GE,<NARGS-2>    .WORD   B
                        .IIF  GE,<NARGS-3>    .WORD   C
                              .ENDM   CMDDEF
                        ;
```

```
                        ;-----------------------------------------------------------------
                        ;  Macro to end a set of table entries.
                        ;
                              .MACRO  TBLEND
                              .CSECT  CMDVS1
                              .WORD   0
                              .CSECT  TSKST1
                              .ENDM   TBLEND
```

```
                        ;------------------------------------------------------------
                        ;  Define options for the SET command.
                        ;
                        ;  Table of "device" names for the SET command and a
                        ;  pointer to a set of sub-options for the device.
                        ;
                        ;
000000                        TBLDEF  SET,3
000002                        CMDDEF  CA*CHE,STCHNB,0,0
000012                        CMDDEF  CC*L,SETSUB,CCLHD,0
000022                        CMDDEF  CO*RTIM,SETSVL,VCORTM,10000.
000032                        CMDDEF  CTRLD,SETSUB,CTDHD,0
000042                        CMDDEF  ED*IT,SETSUB,EDITHD,0
000052                        CMDDEF  EM*T,SETSUB,EMTHD,0
000062                        CMDDEF  ENDS*TARTUP,SETESU,0,0
000072                        CMDDEF  ER*ROR,SETSUB,ERRHD,0
000102                        CMDDEF  EX*IT,RDCMD,0,0
000112                        CMDDEF  HIP*RCT,SETSVL,VHIPCT,10000.
000122                        CMDDEF  HO*ST,SETHST,0,0
000132                        CMDDEF  IND,SETSUB,INDHD,0
000142                        CMDDEF  INT*IOC,SETSVL,VINTIO,10000.
000152                        CMDDEF  IO*ABORT,SETPRV,IOABHD,0
000162                        CMDDEF  KMO*N,SETSUB,KMONHD,0
000172                        CMDDEF  LAN*GUAGE,SETSUB,LANGHD,0
000202                        CMDDEF  LOG,SETLOG,0,0
000212                        CMDDEF  LOGO*FF,SETLOF,0,0
000222                        CMDDEF  MAX*PRIORITY,SETMPR,0,0
000232                        CMDDEF  NUM*DC,SETNDC,NUMDCD,0
000242                        CMDDEF  OFF*TIM,SETSVL,VOFFTM,10000.
000252                        CMDDEF  ONT*IM,SETSVL,VONTM,10000.
000262                        CMDDEF  PHO*NE,SETSVB,VUSPHN,1
000272                        CMDDEF  PROC*ESS,SETPRC,0,0
000302                        CMDDEF  PROM*PT,SETPRT,0,0
000312                        CMDDEF  PRI*ORITY,SETPRI,0,0
000322                        CMDDEF  PRINTS*CREEN,SETPRS,0,0
000332                        CMDDEF  PRINTW*INDOW,SETPRS,0,0
000342                        CMDDEF  QUAN0,SETSVL,VQUAN0,10000.
000352                        CMDDEF  QUAN1,SETSVL,VQUAN1,10000.
000362                        CMDDEF  QUAN1A,SETSVL,VQUN1A,10000.
000372                        CMDDEF  QUAN1B,SETSVL,VQUN1B,10000.
000402                        CMDDEF  QUAN1C,SETSVL,VQUN1C,10000.
000412                        CMDDEF  QUAN2,SETSVL,VQUAN2,10000.
000422                        CMDDEF  QUAN3,SETSVL,VQUAN3,10000.
000432                        CMDDEF  REC*ALL,SETREC,0,0
000442                        CMDDEF  SHUT*DOWN,SETSHT,0,0
000452                        CMDDEF  NOSHUT*DOWN,SETNSH,0,0
000462                        CMDDEF  SIG*NAL,SETSUB,SIGHD,0
000472                        CMDDEF  SL,SETSL,0,0
000502                        CMDDEF  SUB*PROCESS,SETSBP,0,0
000512                        CMDDEF  SYSP*ASSWORD,SETSYP,0,0
000522                        CMDDEF  SYSP*WD,SETSYP,0,0
000532                        CMDDEF  TE*RMINAL,SETTTY,0,0
000542                        CMDDEF  TT*Y,SETTTY,0,0
000552                        CMDDEF  TIMI*N,SETSVL,VTMIN,10000.
000562                        CMDDEF  TIML*OC,SETSVL,VTMLOC,10000.
000572                        CMDDEF  TIMO*UT,SETSVL,VTMOUT,10000.
000602                        CMDDEF  UC*L,SETSUB,UCLHD,0
```

```
      58 000612                                   CMDDEF  US*R,RDCMD,0,0
      59 000622                                   CMDDEF  VER*IFY,STVRFY,0,0
      60 000632                                   CMDDEF  NOVER*IFY,STNOVR,0,0
      61 000642                                   CMDDEF  WI*LDCARDS,SETSUB,WILDHD,0
      62 000652                                   CMDDEF  WIN*DOWS,SETWIN,0,0
      63 000662                                   CMDDEF  WINP*RT,SETPRS,0,0
      64 000672                                   TBLEND
```

```
                                ;
                                ;  Define options for SET EDIT command
                                ;
000000                                TBLDEF   EDIT,3
000676                                CMDDEF   ED*IT,SEEDIT,0
000706                                CMDDEF   KED,SEKED,0
000716                                CMDDEF   K52,SEK52,0
000726                                CMDDEF   T*ECO,SETECO,0
000736                                TBLEND
                                ;
                                ;  Define options for SET ERROR command
                                ;
000000                                TBLDEF   ERR,2
000742                                CMDDEF   W*ARNING,SETSEV,SC$WRN
000750                                CMDDEF   E*RROR,SETSEV,SC$ERR
000756                                CMDDEF   S*EVERE,SETSEV,SC$SEV
000764                                CMDDEF   F*ATAL,SETSEV,SC$FTL
000772                                CMDDEF   U*NCONDITIONAL,SETSEV,SC$UNC
001000                                CMDDEF   N*ONE,SETSEV,SC$NON
001006                                TBLEND
                                ;
                                ;  Define options for SET WILDCARDS command
                                ;
000000                                TBLDEF   WILD,3
001012                                CMDDEF   IM*PLICIT,SETBIT,LSW5,$WILD
001022                                CMDDEF   EX*PLICIT,CLRBIT,LSW5,$WILD
001032                                TBLEND
                                ;
                                ;  Define options for the SET KMON command
                                ;
000000                                TBLDEF   KMON,3
001036                                CMDDEF   IND,SETIND,LSW5,$INDDF
001046                                CMDDEF   NOI*ND,RSTIND,LSW5,$INDDF
001056                                CMDDEF   SY*STEM,CLRBIT,LSW7,$UKMON
001066                                CMDDEF   UCI,SETUKM,0,0
001076                                CMDDEF   USE*R,SETUKM,0,0
001106                                CMDDEF   NEW,SETKNW,0,0
001116                                CMDDEF   DEBUG,SETBIT,LSW9,$DBKMN
001126                                CMDDEF   NODEBUG,CLRBIT,LSW9,$DBKMN
001136                                TBLEND
                                ;
                                ;  Define options for the SET CCL command
                                ;
000000                                TBLDEF   CCL,3
001142                                CMDDEF   T*EST,SETBIT,LSW5,$CLTST
001152                                CMDDEF   NOT*EST,CLRBIT,LSW5,$CLTST
001162                                CMDDEF   NEW,SETCNW,0,0
001172                                TBLEND
                                ;
                                ;  Define options for the SET UCL command
                                ;
000000                                TBLDEF   UCL,3
001176                                CMDDEF   F*IRST,STUCLF,LSW7,$UCLCF
001206                                CMDDEF   M*IDDLE,STUCLM,LSW7,$UCLCM
001216                                CMDDEF   L*AST,STUCLL,LSW7,$UCLCL
001226                                CMDDEF   N*ONE,STUCLN,LSW7,0
001236                                TBLEND
```

```
                                ;
                                ;  Define options for the SET IND command
                                ;
000000                                TBLDEF   IND,3
001242                                CMDDEF   AB*ORT,SETBIT,LSW7,$INDAB
001252                                CMDDEF   NOA*BORT,CLRBIT,LSW7,$INDAB
001262                                TBLEND
                                ;
                                ;  Define options for the SET LD command
                                ;
000000                                TBLDEF   LDOP,1
001266                                CMDDEF   CL*EAN,SLDCLN
001272                                CMDDEF   EMP*TY,SLDEMP
001276                                CMDDEF   FR*EE,SLDFRE
001302                                CMDDEF   WR*ITE,SLDWRT
001306                                CMDDEF   NOWR*ITE,SLDNWR
001312                                TBLEND
                                ;
                                ;  Define options for the SET SL command
                                ;
000000                                TBLDEF   SLE,1
001316                                CMDDEF   AS*K,SLONOP
001322                                CMDDEF   KED,SLOKED
001326                                CMDDEF   K52,SLOKED
001332                                CMDDEF   KEX,SLOKED
001336                                CMDDEF   NOKED,SLORT
001342                                CMDDEF   LEA*RN,SLOUNI
001346                                CMDDEF   NOLEA*RN,SLONOP
001352                                CMDDEF   LET,SLOLET
001356                                CMDDEF   NOLET,SLONLT
001362                                CMDDEF   OF*F,SLOOFF
001366                                CMDDEF   ON,SLOON
001372                                CMDDEF   KM*ON,SLOON
001376                                CMDDEF   RT*11,SLORT
001402                                CMDDEF   SUB*STITUTE,SLOLET
001406                                CMDDEF   NOSUB*STITUTE,SLONLT
001412                                CMDDEF   SYS*GEN,SLONOP
001416                                CMDDEF   TT*YIN,SLOTTY
001422                                CMDDEF   NOTT*YIN,SLONTT
001426                                CMDDEF   WID*TH,SLOWID
001432                                CMDDEF   REC*ALL,SLONOP
001436                                CMDDEF   NOREC*ALL,SLONOP
001442                                TBLEND
                                ;
                                ;  Define options for the SET RECALL
                                ;
000000                                TBLDEF   RCL,1
001446                                CMDDEF   N*ORMAL,0
001452                                CMDDEF   R*EVERSE,1
001456                                TBLEND
                                ;
                                ;  Define options for the SET WINDOW command
                                ;
000000                                TBLDEF   WIN,2
001462                                CMDDEF   COL*UMNS,WINCOL,0
001470                                CMDDEF   DA*RK,WINBIT,WFNORM
001476                                CMDDEF   DEL*ETE,WINBIT,WFDEL
```

```
                                        CMDDEF  LI*GHT,WINBIT,WFREV
                                        CMDDEF  NAR*ROW,WINBIT,WF80
                                        CMDDEF  NOR*MAL,WINBIT,WFNORM
                                        CMDDEF  OFF,WINBIT,WFDEL
                                        CMDDEF  ON,WINBIT,0
                                        CMDDEF  REV*ERSE,WINBIT,WFREV
                                        CMDDEF  SCR*OLL,WINSCR,0
                                        CMDDEF  NOSCR*OLL,WINNOS
                                        CMDDEF  WI*DE,WINBIT,WF132
                                        TBLEND
                                ;
                                ;  Define options for the SET PRINTWINDOW command
                                ;
                                        TBLDEF  PRS,1
                                        CMDDEF  BEL*L,PRSBEL
                                        CMDDEF  NOBEL*L,PRSNBL
                                        CMDDEF  DEV*ICE,PRSDEV
                                        CMDDEF  DR*AFT,PRSDRF
                                        CMDDEF  FL*AG,PRSFLG
                                        CMDDEF  NOFL*AG,PRSNFL
                                        CMDDEF  KEY*PRINT,PRSKEY
                                        CMDDEF  NOKEY*PRINT,PRSNKY
                                        CMDDEF  LET*TERQUALITY,PRSLET
                                        CMDDEF  ST*AMP,PRSSTM
                                        CMDDEF  NOST*AMP,PRSNST
                                        CMDDEF  TY*PE,PRSTYP
                                        CMDDEF  WID*TH,PRSWID
                                        CMDDEF  NOWID*TH,PRSNWD
                                        TBLEND
                                ;
                                ;  Printer types for SET PRINTWINDOW/TYPE=type
                                ;
                                        TBLDEF  PWT,2
                                        CMDDEF  AS*CII,0,0
                                        CMDDEF  FOR*EIGN,0,0
                                        CMDDEF  PRINT*RONIX,0,0
                                        CMDDEF  SIM*PLE,0,0
                                        CMDDEF  ST*ANDARD,0,0
                                        CMDDEF  LA36,0,0
                                        CMDDEF  LA120,0,0
                                        CMDDEF  LA100,1,PA$GRC!PA$UKC!PA$ULN!PA$HQL!PA$DWD
                                        CMDDEF  LA12,1,PA$GRC!PA$UKC!PA$BLD!PA$ULN!PA$HQL!PA$DWD
                                        CMDDEF  LA50,1,PA$GRC!PA$UKC!PA$DSC!PA$BLD!PA$ULN!PA$HQL!PA$DWD
                                        CMDDEF  LA210,1,PA$GRC!PA$UKC!PA$DSC!PA$BLD!PA$ULN!PA$HQL!PA$DWD
                                        CMDDEF  LQP02,1,PA$BLD!PA$ULN
                                        CMDDEF  LQP03,1,PA$BLD!PA$ULN
                                        CMDDEF  LN03,1,PA$GRC!PA$BLD!PA$ULN!PA$DWD
                                        TBLEND
                                ;
                                ;  Define options for the SET LANGUAGE command
                                ;
                                        TBLDEF  LANG,3
                                        CMDDEF  DI*BOL,SETBIT,LSW6,$DIBOL
                                        CMDDEF  DBL,CLRBIT,LSW6,$DIBOL
                                        TBLEND
                                ;
                                ;  Define options for the SET EMT command
```

```
                                ;
000000                                  TBLDEF  EMT,3
002046                                  CMDDEF  TR*ACE,SETEMT,LSW6,$EMTTR
002056                                  CMDDEF  NOTR*ACE,CLRBIT,LSW6,$EMTTR
002066                                  TBLEND
                                ;
                                ;  Define options for the SET IOABORT command
                                ;
000000                                  TBLDEF  IOAB,3
002072                                  CMDDEF  AB*ORT,STRWRD,IOABFL,1
002102                                  CMDDEF  NOAB*ORT,STRWRD,IOABFL,0
002112                                  TBLEND
                                ;
                                ;  Define options for the SET SIGNAL command
                                ;
000000                                  TBLDEF  SIG,3
002116                                  CMDDEF  H*IPRCT,SETBIT,LSW8,$SGHIO
002126                                  CMDDEF  NOH*IPRCT,CLRBIT,LSW8,$SGHIO
002136                                  CMDDEF  I*NTIOC,SETBIT,LSW8,$SGIIO
002146                                  CMDDEF  NOI*NTIOC,CLRBIT,LSW8,$SGIIO
002156                                  CMDDEF  QUANO,SETBIT,LSW8,$SGQO
002166                                  CMDDEF  NOQUANO,CLRBIT,LSW8,$SGQO
002176                                  CMDDEF  QUAN1,SETBIT,LSW8,$SGQ1
002206                                  CMDDEF  NOQUAN1,CLRBIT,LSW8,$SGQ1
002216                                  CMDDEF  QUAN1A,SETBIT,LSW8,$SGQ1A
002226                                  CMDDEF  NOQUAN1A,CLRBIT,LSW8,$SGQ1A
002236                                  CMDDEF  QUAN1B,SETBIT,LSW8,$SGQ1B
002246                                  CMDDEF  NOQUAN1B,CLRBIT,LSW8,$SGQ1B
002256                                  CMDDEF  QUAN1C,SETBIT,LSW8,$SGQ1C
002266                                  CMDDEF  NOQUAN1C,CLRBIT,LSW8,$SGQ1C
002276                                  CMDDEF  QUAN2,SETBIT,LSW8,$SGQ2
002306                                  CMDDEF  NOQUAN2,CLRBIT,LSW8,$SGQ2
002316                                  CMDDEF  QUAN3,SETBIT,LSW8,$SGQ3
002326                                  CMDDEF  NOQUAN3,CLRBIT,LSW8,$SGQ3
002336                                  CMDDEF  OFF,CLRBIT,LSW8,$SGALL
002346                                  TBLEND
                                ;
                                ;  Define options for the SET LOG command
                                ;
000000                                  TBLDEF  STLG,1
002352                                  CMDDEF  A*LL,STLGAL
002356                                  CMDDEF  CLE*AN,STLGCN
002362                                  CMDDEF  CLO*SED,STLGCL
002366                                  CMDDEF  FI*LE,STLGFL
002372                                  CMDDEF  I*NPUT,STLGIN
002376                                  CMDDEF  O*UTPUT,STLGOT
002402                                  CMDDEF  WR*ITE,STLGWR
002406                                  CMDDEF  NOWR*ITE,STLGNW
002412                                  TBLEND
                                ;
                                ;  Define options for the SET LOGOFF command
                                ;
000000                                  TBLDEF  STLO,1
002416                                  CMDDEF  FI*LE,STLOFL
002422                                  TBLEND
                                ;
                                ;  Define options for the SET SUBPROCESS command
```

```
                                        ;
000000                                          TBLDEF  SBP,1
002426                                          CMDDEF  FI*LE,SBPFIL
002432                                          TBLEND
                                        ;
                                        ;  Define options for the SET CTRLD command
                                        ;
000000                                          TBLDEF  CTD,1
002436                                          CMDDEF  D*EBUG,STCDON
002442                                          CMDDEF  NOD*EBUG,STCDOF
002446                                          TBLEND
```

```
                                ;-------------------------------------------------------------------------------
                                ;  Data areas
                                ;
000000                          HANSAV: .BLKW   5               ;Space for .SAVESTATUS
000012  012276                  R50CLO: .RAD50  /CLO/
000014  013666                  R50C10: .RAD50  /C10/
000016  000000                  WINFLG: .WORD   0
                                ;
                                ;  Flags stored in WINFLG
                                ;
        000001                  WFDEL   =       1               ;Delete the window
        000002                  WFREV   =       2               ;Put terminal in reverse video
        000004                  WFNORM  =       4               ;Put terminal in normal video mode
        000010                  WF132   =       10              ;Put terminal in 132 column mode
        000020                  WF80    =       20              ;Put terminal in 80 column mode
                                ;
                                ;  Terminal control sequences
                                ;
000020   033   133   077        TCREV:  .ASCII  <33>/[?5h/<200> ;Select reverse video display
000023   065   150   200
000026   033   133   077        TCNORM: .ASCII  <33>/[?5l/<200> ;Select normal video display
000031   065   154   200
000034   033   133   077        TC132:  .ASCII  <33>/[?3h/<200> ;Select 132 column mode
000037   063   150   200
000042   033   133   077        TC80:   .ASCII  <33>/[?3l/<200> ;Select 80 column mode
000045   063   154   200
                                        .EVEN
                                ;
                                ;  Emt to set line speed
                                ;
000050   000   154              LSPEMT: .BYTE   0,154
000052  000000                          .WORD   0               ;Line number
000054  000000                          .WORD   0               ;Speed code
                                ;
                                ;  Emt to reinitialize data caching
                                ;
000056   000   126              DCHNEW: .BYTE   0,126
000060  000006                          .WORD   6
                                ;
                                ;  Emt to set job priority
                                ;
000062   000   150              PRIEMT: .BYTE   0,150
000064  000000                          .WORD   0
                                ;
                                ;  Emt to create window # 1
                                ;
000066   000   161              WINMAK: .BYTE   0,161
000070   001   001                      .BYTE   1,1
000072   120   020                      .BYTE   80.,16.
000074  000000  000000                   .WORD   0,0
                                ;
                                ;  Emt to map to window # 1
                                ;
000100   001   161              WINMAP: .BYTE   1,161
000102   001   000                      .BYTE   1,0
                                ;
                                ;  Emt to delete window # 1
```

```
                                        ;
000104      002      161        WINDEL: .BYTE   2,161
000106      001      000                .BYTE   1,0
                                        ;
                                        ;  Emt to dismount all LD's
                                        ;
000110      005      135        DMTALD: .BYTE   5,135
000112   000000                         .WORD   0
```

```
                                        .SBTTL  SET command
                                ;-----------------------------------------------------------------------
                                ;  Process the "SET" command.
                                ;
 000114  004767  000000G        CMDSET: CALL    CVTTAB          ;CONVERT TAB AND FF CHARS TO SPACES
 000120  004767  000000G                CALL    SKPSPC          ;Skip up to start of device name
 000124  010305                         MOV     R3,R5           ;SAVE COMMAND STRING POINTER
 000126  105067  000000G                CLRB    NOFLAG          ;ASSUME "NO" OPTION NOT SEEN YET
                                ;
                                ;  If device name ends with a colon (e.g., "TT:") replace the colon
                                ;  with a space.
                                ;
 000132  112300                 7$:     MOVB    (R3)+,R0        ;Get next char of device name
 000134  120027  000040                 CMPB    R0,#40          ;Reached end of name?
 000140  101410                         BLOS    8$              ;Br if yes
 000142  120027  000057                 CMPB    R0,#'/          ;Start of qualifiers?
 000146  001405                         BEQ     8$              ;Br if yes
 000150  120027  000072                 CMPB    R0,#':          ;Is this character colon?
 000154  001366                         BNE     7$              ;Br if not
 000156  112743  000040                 MOVB    #40,-(R3)       ;Replace colon with space
 000162  010503                 8$:     MOV     R5,R3           ;Get back pointer to start of device name
                                ;
                                ;  Look up device name and see if we recognize it as a pseudo-device
                                ;  such as TT, WILDCARD, etc.
                                ;
 000164  012704  000000'                MOV     #SETHD,R4       ;GET TABLE OF PSEUDO-DEVICE NAMES
 000170  004767  000000G                CALL    SEARCH          ;SEARCH FOR DEVICE NAME
 000174  103401                         BCS     1$              ;BR IF NOT A PSEUDO-DEVICE NAME
                                ;
                                ;  This is a pseudo-device name.
                                ;  Jump off to appropriate processing routine.
                                ;
 000176  000134                         JMP     @(R4)+          ;JUMP OFF TO PROCESSING ROUTINE
                                ;
                                ;  Device name is not recognized as a pseudo device.
                                ;  Apply any logical device assignment to the device name.
                                ;
 000200  010503                 1$:     MOV     R5,R3           ;POINT BACK TO DEVICE NAME
 000202  005067  000002G                CLR     R50BUF+2        ;CLEAR 2ND WORD OF RAD50 NAME
 000206  004767  000000G                CALL    GTRD50          ;ACCRUE RAD50 VALUE
 000212  016700  000000G                MOV     R50BUF,R0       ;GET DEVICE NAME
 000216  005767  000002G                TST     R50BUF+2        ;MAKE SURE NAME ONLY 3 CHARS LONG
 000222  001404                         BEQ     2$              ;BR IF OK
 000224                                 FABORT  #CSIMS2         ;INVALID DEVICE FOR SET
 000234  004767  000000G        2$:     CALL    ASNSRC          ;Search for device name in assign table
 000240  103404                         BCS     6$              ;Br if name was not assigned
 000242  016200  000000G                MOV     AT$DEV(R2),R0   ;Get the physical device name
 000246  010067  000000G                MOV     R0,R50BUF       ;Set new device name for SET
                                ;
                                ;  See if this is a SET LDn command
                                ;
 000252  020067  000000G        6$:     CMP     R0,R50LD        ;SEE IF IT IS LD
 000256  103410                         BLO     3$              ;BR IF NOT
 000260  020067  000000G                CMP     R0,R50LD7       ;CHECK HIGH RANGE
 000264  101005                         BHI     3$              ;BR IF NOT AN LD UNIT
 000266  105767  000000G                TSTB    VLDSYS          ;IS LD SUPPORT GENNED IN?
 000272  001416                         BEQ     5$              ;BR IF NOT, maybe using an actual LD.TSX
```

```
     58 000274  000167  005560                    JMP     SETLD           ;GO PROCESS SET LD COMMAND
     59                                     ;
     60                                     ;  See if this is a SET CLn command
     61                                     ;  Note, that CL unit allocation is now checked only for some SET commands.
     62                                     ;
     63 000300  005727  000000G             3$:     TST     #CLTOTL         ;Are there any CL lines?
     64 000304  001411                              BEQ     5$              ;Br if not
     65 000306  004767  000000G                     CALL    CHKCLU          ;See if this is a CL or C1 unit (unit # to R0)
     66 000312  103003                              BCC     4$              ;BR if it is a valid CL or C1 unit
     67 000314  016700  000000G                     MOV     R50BUF,R0       ;If not, we need to get back RAD50 device name
     68 000320  000403                              BR      5$              ;Try it as a "real" device
     69 000322  010002                      4$:     MOV     R0,R2           ;Pass CL unit number in R2
     70 000324  000167  000000G                     JMP     SETCL           ;Process SET CL command
     71                                     ;
     72                                     ;  See if this device is allocated to another user before trying
     73                                     ;  to perform handler set code.
     74                                     ;
     75 000330  004767  000000G             5$:     CALL    CHKALC          ;Check for allocation conflict
     76 000334  000167  000470                      JMP     SETDEV
```

```
                                ;-------------------------------------------------------------------------------
                                ;  Process SET commands that have sub-options
                                ;
                                ;  Inputs:
                                ;   R3 = Pointer to start of qualifier in command line.
                                ;   R4 = Points to start of command table built with TBLDEF...TBLEND macros.
                                ;
                                ;  Outputs:
                                ;   R3 = Pointer beyond end of qualifier.
                                ;   R4 = Pointer to argument value word in command table.
                                ;
                                ;  See if set option is preceded by "NO".
                                ;
000340  004767  0000000G      SETPRV: CALL    CKSYPV          ;Must have SYSPRV privilege
000344  010246                SETSUB: MOV     R2,-(SP)
000346  010546                        MOV     R5,-(SP)
000350  122327  000072                CMPB    (R3)+,#':       ;WAS DEVICE NAME FOLLOWED BY COLON?
000354  001401                        BEQ     3$              ;BR IF YES
000356  005303                        DEC     R3              ;RESET POINTER IF NOT
000360  004767  0000000G      3$:     CALL    SKPSPC          ;SKIP OVER SPACES
000364  111300                        MOVB    (R3),R0         ;Get next character from command
000366  120027  000075                CMPB    R0,#'=          ;Is there an equal sign?
000372  001403                        BEQ     5$              ;Br if yes -- skip over equal sign
000374  120027  000072                CMPB    R0,#':          ;Colon separator?
000400  001001                        BNE     6$              ;Br if not
000402  005203                4$:     INC     R3              ;Skip over separator
000404  010302                6$:     MOV     R3,R2           ;SAVE COMMAND STRING POINTER
000406  004767  0000000G              CALL    GTRD50          ;ACCRUE NEXT WORD IN RAD50 FORM
000412  026767  0000000G 0000000G      CMP     R50BUF,R50NO    ;IS WORD "NO"?
000420  001005                        BNE     1$              ;BR IF NOT
                                ;
                                ;  Option is preceeded by NO.  Concatenate NO with following qualifier.
                                ;
000422  010305                        MOV     R3,R5           ;GET POINTER INTO COMMAND PAST "NO"
000424  004767  0000000G              CALL    SKPSPC          ;SKIP OVER SPACES FOLLOWING "NO"
000430  112325                4$:     MOVB    (R3)+,(R5)+     ;CONCATENATE OPTION WORD WITH "NO"
000432  001376                        BNE     4$              ;MOVE ALL OF COMMAND
000434  010203                1$:     MOV     R2,R3           ;GET BACK POINTER TO OPTION WORD
                                ;
                                ;  Look up the option word.
                                ;
000436  011404                        MOV     @R4,R4          ;GET POINTER TO TABLE OF OPTIONS FOR COMMAND
000440  004767  0000000G              CALL    SEARCH          ;LOOK FOR THE CORRECT OPTION WORD
000444  103403                        BCS     BDSO            ;BR IF INVALID OPTION
                                ;
                                ;  Enter Option processing routine.
                                ;
000446  012605                        MOV     (SP)+,R5
000450  012602                        MOV     (SP)+,R2
000452  000134                        JMP     @(R4)+          ;PERFORM PROCESSING FOR OPTION
                                ;
                                ;  Invalid option.
                                ;
000454  005704                BDSO:   TST     R4              ;INVALID OR AMBIGUOUS OPTION?
000456  001407                        BEQ     INVSOP          ;BR IF INVALID
000460                                  FABORT  #AMBOPT         ;AMBIGUOUS OPTION
000470                          INVLDO: .PURGE  #HANCHN         ;MAKE SURE THE CHANNEL IS CLOSED
```

```
     58 000476                          INVSOP:  FABORT  #INVOPT          ;INVALID OPTION
```

```
                                        .SBTTL  .       System parameters
                                ;-------------------------------------------------------------------
                                ;  Set a system parameter value.
                                ;
                                ;  Inputs:
                                ;   R4 Points to cell with address of parameter cell to be modified,
                                ;   following word contains maximum legal value for parameter.
                                ;
000506  012746  177777          SETSVB: MOV     #-1,-(SP)       ;Signal byte value
000512  000401                          BR      SETSVC          ;Br to common code
000514  005046                  SETSVL: CLR     -(SP)           ;Signal word value
000516  004767  0000000G        SETSVC: CALL    CKSYPV          ;Must have SYSPRV privilege
                                ;
                                ;  Accrue numeric value
                                ;
000522  004767  0000000G                CALL    ACRDEC          ;ACCRUE A DECIMAL VALUE
000526  016400  000002                  MOV     2(R4),R0        ;GET MAX LEGAL VALUE FOR PARAMETER
                                ;
                                ;  Make sure the value is valid.
                                ;  R1 = Accrued value,  R0 = Maximum legal value.
                                ;
000532  005701                  SETSVV: TST     R1              ;CHECK FOR NEGATIVE VALUE SPECIFIED
000534  002402                          BLT     3$              ;BR IF INVALID VALUE SPECIFIED
000536  020100                          CMP     R1,R0           ;IS SPECIFIED VALUE OK?
000540  101413                          BLOS    1$              ;BR IF OK
000542  010001                  3$:     MOV     R0,R1           ;GET MAX LEGAL VALUE
000544  010005                          MOV     R0,R5
000546                                  .PRINT  #MAXAVL         ;VALUE TOO LARGE
000554  004767  0000000G                CALL    PRTDEC          ;PRINT MAX LEGAL VALUE
000560                                  .PRINT  #CRLF           ;END LINE
000566  000406                          BR      4$              ;FINISHED
                                ;
                                ;  Store value into parameter cell
                                ;
000570  005726                  1$:     TST     (SP)+           ;Word or byte value?
000572  001403                          BEQ     2$              ;Br if word
000574  110174  000000                  MOVB    R1,@(R4)        ;Store byte value
000600  000401                          BR      4$              ;Finish
000602  010134                  2$:     MOV     R1,@(R4)+       ;STORE NEW VALUE INTO PARAMETER CELL
000604  000167  0000000G        4$:     JMP     RDCMD           ;FINISHED
                                ;
                                ;  SET NUMDC value
                                ;
000610  004767  0000000G        SETNDC: CALL    CKSYPV          ;Must have SYSPRV privilege
                                ;
                                ;  Accrue numeric value
                                ;
000614  004767  0000000G                CALL    ACRDEC          ;ACCRUE A DECIMAL VALUE
000620  016700  0000000G                MOV     VNUMDC,R0       ;GET MAX LEGAL VALUE FOR PARAMETER
000624  000742                          BR      SETSVV          ;Check and store value
```

```
                                        ;
                                        ;  SET CACHE = value
                                        ;
000626  004767  000000G          STCHNB: CALL    CKSYPV          ;Must have SYSPRV privilege
                                        ;
                                        ;  Accrue numeric value
                                        ;
000632  004767  000000G                 CALL    ACRDEC          ;ACCRUE A DECIMAL VALUE
000636  016700  000000G                 MOV     CSHALC,R0       ;GET MAX LEGAL VALUE FOR PARAMETER
000642  005701                          TST     R1              ;CHECK FOR NEGATIVE VALUE SPECIFIED
000644  002402                          BLT     3$              ;BR IF INVALID VALUE SPECIFIED
000646  020100                          CMP     R1,R0           ;IS SPECIFIED VALUE OK?
000650  101412                          BLOS    1$              ;BR IF OK
000652  010001                  3$:     MOV     R0,R1           ;GET MAX LEGAL VALUE
000654  010005                          MOV     R0,R5
000656                                  .PRINT  #MAXAVL         ;VALUE TOO LARGE
000664  004767  000000G                 CALL    PRTDEC          ;PRINT MAX LEGAL VALUE
000670                                  .PRINT  #CRLF           ;END LINE
                                        ;
                                        ;  Store value into parameter cell
                                        ;
000676  010167  000000G          1$:    MOV     R1,VCSHNB       ;STORE NEW VALUE INTO PARAMETER CELL
                                        ;
                                        ;  Now clean out the data cache
                                        ;
000702  012700  000056'                 MOV     #DCHNEW,R0      ;EMT to clean out the cache
000706  104375                          EMT     375
000710  000167  000000G                 JMP     RDCMD           ;FINISHED
                                        ;
                                        ;  SET CTRLD [NO]DEBUG
                                        ;
000714  105767  000000G          STCDON: TSTB    VDBFLG          ;Was debugger genned into the system?
000720  001004                          BNE     1$              ;Br if yes
000722                                  FABORT  #EM$NPD         ;No program debugger
000732  032767  000000G 000000G  1$:     BIT     #PO$DBG,PRIVCO  ;Can we use debugger?
000740  001004                          BNE     2$              ;Br if yes
000742                                  FABORT  #EM$NAD         ;Not allowed to use debugger
000752  052761  000000G 000000G  2$:     BIS     #$CTRLD,LSW9(R1);Enable ctrl-D debugger entry
000760  000167  000000G                 JMP     RDCMD
000764  042761  000000G 000000G STCDOF: BIC     #$CTRLD,LSW9(R1);Disable ctrl-D debugger entry
000772  000167  000000G                 JMP     RDCMD
                                        ;
                                        ;  SET EMT TRACE
                                        ;
000776  032767  000000G 000000G SETEMT: BIT     #PO$DBG,PRIVCO  ;Are we allowed to use debugger?
001004  001004                          BNE     1$              ;Br if yes
001006                                  FABORT  #EM$NAD         ;Not allowed to use debugger
001016  052761  000000G 000000G  1$:     BIS     #$EMTTR,LSW6(R1);Enable EMT trace
001024  000167  000000G                 JMP     RDCMD
```

```
                                        .SBTTL  .       Real device
                                ;-------------------------------------------------------------------------
                                ;  The device name is not a pseudo-device.
                                ;  See if it is the name of a real device.
                                ;
001030  016701  000000G        SETDEV: MOV     R50BUF,R1       ;GET THE DEVICE NAME AND UNIT
001034  005000                        CLR     R0              ;CLEAR HIGH ORDER REGISTER
001036  071027  000050                DIV     #50,R0          ;STRIP THE LAST RAD 50 CHARACTER
001042  010146                        MOV     R1,-(SP)        ;SAVE THE UNIT NUMBER
001044  070027  000050                MUL     #50,R0          ;GET THE DEVICE NAME SHIFTED
001050  010167  000000G               MOV     R1,R50BUF       ;AND STORE BACK INTO THE RAD 50 BUFFER
001054  012601                        MOV     (SP)+,R1        ;GET THE UNIT NUMBER
001056  001407                        BEQ     2$              ;BR IF NO UNIT NUMBER WAS SPECIFIED
001060  162701  000036                SUB     #36,R1          ;NORMALIZE TO ZERO
001064  002450                        BLT     SETIVD          ;ERROR IF LESS THAN ZERO
001066  020127  000007                CMP     R1,#7           ;CHECK UNIT RANGE
001072  003045                        BGT     SETIVD          ;ERROR IF GREATER THAN SEVEN
001074  000402                        BR      3$              ;CONTINUE
001076  012701  100000        2$:     MOV     #100000,R1      ;SET NO UNIT WAS SPECIFIED FLAG
001102  010167  000000G       3$:     MOV     R1,DEVUNT       ;SAVE THE UNIT SPECIFIED
001106  122327  000072                CMPB    (R3)+,#':       ;WAS DEVICE NAME FOLLOWED BY COLON?
001112  001401                        BEQ     4$              ;BR IF YES
001114  005303                        DEC     R3              ;BACKUP POINTER
                                ;
                                ;  Lookup the handler file and read in the first 2 blocks.
                                ;
001116  016767  000000G 000002G 4$:     MOV     R50BUF,HNBUF+2  ;SET UP HANDLER NAME
001124                                  .LOOKUP #XAREA,#HANCHN,#HNBUF;TRY TO LOOKUP HANDLER FILE
001144  103420                        BCS     SETIVD          ;BR IF CAN'T FIND HANDLER
001146                                  .READW  #XAREA,#HANCHN,#BLKO,#512.,#0 ;READ IN 1ST 2 BLOCKS
001204  103007                        BCC     SETCKP          ;Branch if read ok
                                ;
                                ;  Invalid device name.
                                ;
001206                          SETIVD: .PURGE  #HANCHN         ;MAKE SURE CHANNEL IS CLOSED
001214                                  FABORT  #CSIMS2         ;INVALID DEVICE NAME
                                ;
                                ;  We are performing a set to a device handler.
                                ;  User must be privileged to do this.
                                ;
001224                          SETCKP: .SAVEST #XAREA,#HANCHN,#HANSAV ;Save channel status in HANSAV
001244  032767  000000G 000000G        BIT     #PO$OPR,PRIVCO  ;Is user privileged?
001252  001007                        BNE     NXTOPT          ;Br if yes
001254                                  .PURGE  #HANCHN         ;Close handler channel
001262                                  FABORT  #EM$OPR         ;Not-privileged error message
```

```
                                        ;
                                        ;  Accrue the set option word.
                                        ;
001272  105067  000000G       NXTOPT:  CLRB    NOFLAG          ;RESET THE "NO" OPTION FLAG
001276  004767  000000G                CALL    SKPSPC          ;SKIP OVER ANY SPACES
                                        ;  Check for "NO" preceding the option word.
001302  010302                         MOV     R3,R2           ;SAVE COMMAND STRING POINTER
001304  004767  000000G                CALL    GETKCH          ;GET NEXT COMMAND CHARACTER
001310  120027  000116                 CMPB    R0,#'N          ;IS IT "N"?
001314  001012                         BNE     7$              ;BR IF NOT
001316  004767  000000G                CALL    GETKCH          ;CHECK NEXT CHARACTER
001322  120027  000117                 CMPB    R0,#'O          ;IS IT "O"?
001326  001005                         BNE     7$              ;BR IF NOT
001330  105267  000000G                INCB    NOFLAG          ;REMEMBER "NO" SPECIFIED
001334  004767  000000G                CALL    SKPSPC          ;SKIP OVER SPACES FOLLOWING "NO"
001340  000401                         BR      9$
001342  010203                 7$:     MOV     R2,R3           ;RESET POINTER TO START OF OPTION WORD
                                        ;  Accrue the option word.
001344  004767  000000G        9$:     CALL    GTRD50          ;ACCRUE AS A RAD50 VALUE
                                        ;
                                        ;  Lookup the option word in the handler header.
                                        ;
001350  012704  000400G        SETOPT:  MOV     #BLK0+400,R4    ;POINT TO START OF HANDLER TABLE OF SET OPTIONS
001354  005764  000000G        10$:     TST     SH$VAL(R4)      ;HAVE WE REACHED THE END OF THE TABLE?
001360  001002                         BNE     11$             ;CONTINUE CHECKING
001362  000167  177102                 JMP     INVLDO          ;BR IF YES -- CAN'T FIND OPTION WORD
001366  026764  000000G 000000G 11$:     CMP     R50BUF,SH$NAM(R4);COMPARE NAME WITH THIS ENTRY
001374  001004                         BNE     2$              ;BR IF MISMATCH
001376  026764  000002G 000002G        CMP     R50BUF+2,SH$NAM+2(R4)
001404  001403                         BEQ     9$              ;BR IF FOUND ENTRY FOR THIS OPTION
001406  062704  000000G         2$:     ADD     #SH$$SZ,R4      ;POINT TO NEXT OPTION ENTRY
001412  000760                         BR      10$             ;CONTINUE SEARCH
                                        ;
                                        ;  Found entry for this option.
                                        ;
001414  116402  000000G        9$:     MOVB    SH$RTN(R4),R2   ;GET OFFSET TO ROUTINE TO CALL
001420  042702  177400                 BIC     #^C377,R2       ;CLEAR SIGN EXTENSION
001424  006302                         ASL     R2              ;CONVERT FROM WORD TO BYTE OFFSET
001426  062702  000400G                ADD     #BLK0+400,R2    ;ADD BASE ADDRESS
                                        ;  See if "NO" option is allowed.
001432  105767  000000G                TSTB    NOFLAG          ;WAS "NO" SPECIFIED WITH OPTION?
001436  001410                         BEQ     3$              ;BR IF NOT
001440  132764  000000G 000000G        BITB    #SO$NO,SH$FLG(R4);IS NO ALLOWED WITH THIS OPTION?
001446  001002                         BNE     14$             ;BR IF YES
001450  000167  000464                 JMP     SETIVS          ;INVALID OPTION
001454  062702  000004         14$:    ADD     #4,R2           ;GET ADDRESS OF SET ROUTINE FOR NO-OPTION
                                        ;  See if option takes a numeric parameter.
001460  132764  000000G 000000G 3$:     BITB    #SO$NVL,SH$FLG(R4);DOES OPTION REQUIRE A NUMERIC PARAMETER?
001466  001433                         BEQ     4$              ;BR IF NOT
001470  004767  000000G                CALL    SKPSPC          ;SKIP OVER ANY SPACES
001474  122327  000075                 CMPB    (R3)+,#'=       ;IS OPTION WORD FOLLOWED BY EQUAL SIGN?
001500  001401                         BEQ     7$              ;BR IF YES
001502  005303                         DEC     R3              ;BACKUP POINTER
001504  004767  000000G         7$:     CALL    SKPSPC          ;SKIP OVER ANY SPACES
001510  010305                         MOV     R3,R5           ;SAVE POINTER TO START OF NUMBER
001512  132764  000000G 000000G        BITB    #SO$OCT,SH$FLG(R4);IS NUMERIC VALUE OCTAL?
001520  001003                         BNE     6$              ;BR IF YES
```

```
      58 001522  004767  000000G                 CALL    ACRDEC          ;ACCRUE A DECIMAL VALUE
      59 001526  000402                          BR      5$
      60 001530  004767  000000G           6$:   CALL    ACROCT          ;ACCRUE AN OCTAL VALUE
      61 001534  010100                    5$:   MOV     R1,R0           ;GET VALUE IN R0 FOR SET ROUTINE
      62 001536  020305                          CMP     R3,R5           ;DID WE ACTUALLY GET A NUMBER?
      63 001540  001002                          BNE     15$             ;BR IF YES
      64 001542  000167  000372                  JMP     SETIVS          ;BR IF NOT
      65 001546  122327  000056           15$:   CMPB    (R3)+,#'.       ;WAS DECIMAL POINT SPECIFIED?
      66 001552  001401                          BEQ     4$              ;BR IF YES
      67 001554  005303                          DEC     R3              ;BACKUP POINTER
      68                                ;
      69                                ;  Open channel 17 (which is normally used for KMON overlays) to
      70                                ;  the handler file so that some handlers can use this channel to
      71                                ;  read in and updated additional blocks in the handler.
      72                                ;
      73 001556  010046                    4$:   MOV     R0,-(SP)        ;Save numeric parameter value
      74 001560                                   .PURGE  #17             ;Purge channel 17 (KMON overlay channel)
      75 001566                                   .REOPEN #XAREA,#17,#HANSAV ;Open channel 17 to handler file
      76 001606  012600                           MOV     (SP)+,R0        ;Restore numeric parameter value
      77                                ;
      78                                ;  Call routine in handler that actually does the set.
      79                                ;  On entry the following registers are set up:
      80                                ;   R0 = Optional numeric parameter value.
      81                                ;   R3 = Fixed option value found in option word (SH$VAL)
      82                                ;
      83 001610  010346                          MOV     R3,-(SP)        ;SAVE COMMAND STRING POINTER
      84 001612  016403  000000G                 MOV     SH$VAL(R4),R3   ;GET VALUE FROM OPTION ENTRY
      85 001616  016701  000000G                 MOV     DEVUNT,R1       ;GET THE DEVICE UNIT SPECIFIED
      86 001622  000241                          CLC                     ;CLEAR C-BIT BEFORE ENTERING HANDLER
      87 001624  004712                          CALL    @R2             ;CALL SET PROCESSING ROUTINE IN HANDLER
      88 001626  012603                          MOV     (SP)+,R3        ;RECOVER COMMAND STRING POINTER
      89 001630  012702  000000                  MOV     #0,R2           ;SAY NO ERROR (DON'T AFFECT C-FLAG)
      90 001634  103001                          BCC     13$             ;BR IF NO ERROR OCCURRED
      91 001636  005202                          INC     R2              ;SET ERROR FLAG IN R2
      92                                ;
      93                                ;  Restore status of HANCHN and channel 17
      94                                ;
      95 001640                         13$:      .PURGE  #17             ;Purge channel 17
      96 001646                                   .REOPEN #XAREA,#17,#KMNCHN ;Reopen channel 17 to KMON
      97 001666                                   .REOPEN #XAREA,#HANCHN,#HANSAV ;Reopen HANCHN to handler
      98 001706  005702                          TST     R2              ;Any error detected in handler?
      99 001710  001113                          BNE     SETIVS          ;Br if yes
     100                                ;
     101                                ;  See if there are more options.
     102                                ;
     103 001712  004767  000000G                 CALL    SKPSPC          ;SKIP SPACES
     104 001716  112300                          MOVB    (R3)+,R0        ;GET NEXT CHAR
     105 001720  001406                          BEQ     SETUP           ;BR IF REACHED END
     106 001722  120027  000054                  CMPB    R0,#',          ;IS COMMA THE DELIMITER?
     107 001726  001401                          BEQ     12$             ;BR IF YES
     108 001730  005303                          DEC     R3              ;BACKUP STRING POINTER
     109 001732  000167  177334           12$:   JMP     NXTOPT          ;GO PROCESS THE NEXT OPTION
```

```
                                        ;
                                        ;  Finished with all options.
                                        ;  Write updated handler back to its disk file.
                                        ;
001736                                  SETUP:  .WRITW  #XAREA,#HANCHN,#BLK0,#512.,#0
001774                                          .CLOSE  #HANCHN
                                        ;
                                        ;  Now try to update the running handler in memory.
                                        ;
002002  016702  000000G                         MOV     NUMDEV,R2       ;GET INDEX TO LAST DEVICE IN DEVICE TABLES
002006  016700  000002G                         MOV     HNBUF+2,R0      ;Get device name
002012  020062  000000G                 1$:     CMP     R0,PNAME(R2)    ;LOOKUP DEVICE NAME
002016  001412                                  BEQ     3$              ;BR IF FOUND
002020  162702  000002                          SUB     #2,R2           ;MORE ENTRIES TO CHECK?
002024  003372                                  BGT     1$              ;BR IF YES
002026                                          FWARN   #EM$HNI         ;Warn that handler is not installed
002042  000434                                  BR      2$
                                        ;
                                        ;  We have located the entry for the handler.
                                        ;  See if we are allowed to change the in-memory handler image.
                                        ;
002044  032762  000000G 000000G         3$:     BIT     #DX$NST,DVFLAG(R2);Are we allowed to reload running handler?
002052  001407                                  BEQ     4$              ;Br if yes
002054                                          FWARN   #TM$MRS         ;Tell user that he must reboot the system
002070  000421                                  BR      2$
002072  012700  000000G                 4$:     MOV     #HUPARG,R0      ;POINT TO ARG BLOCK FOR UPDATE INFO
002076  010260  000004                          MOV     R2,4(R0)        ;SET DEVICE INDEX NUMBER
002102  005067  001006G                         CLR     BLK0+1006       ;Clear LQE in handler image
002106  005067  001010G                         CLR     BLK0+1010       ;Clear CQE in handler image
002112  012760  001000G 000006                  MOV     #BLK0+1000,6(R0);SET ADDRESS OF START OF NEW HANDLER CODE
002120  104375                                  EMT     375             ;TRY TO UPDATE RUNNING HANDLER
002122  103004                                  BCC     2$              ;BR IF SUCCESSFUL
002124                                          FABORT  #HANBSY         ;HANDLER WAS ACTIVE
002134  000167  000000G                 2$:     JMP     RDCMD           ;FINISHED
                                        ;
                                        ;  Invalid command
                                        ;
002140                                  SETIVS: .PURGE  #HANCHN         ;MAKE SURE CHANNEL IS CLOSED
002146                                          FABORT  #CSIMS1
```

```
                                        ;
                                        ;  Process options which only involve setting/resetting flags.
                                        ;
                                        ;  Set a bit in a user table
                                        ;
002156  004767  0000000G        PRVSET: CALL    CKPRIV          ;REQUIRE PRIVILEGE
002162  012405                SETBIT: MOV     (R4)+,R5        ;POINT TO TABLE
002164  060105                        ADD     R1,R5           ;POINT TO TABLE ENTRY FOR THIS USER
002166  051415                        BIS     @R4,@R5         ;SET THE DESIRED FLAG
002170  000167  0000000G              JMP     RDCMD
                                        ;
                                        ;  Reset a bit in a user table
                                        ;
002174  004767  0000000G        PRVCLR: CALL    CKPRIV          ;REQUIRE PRIVILEGE
002200  012405                CLRBIT: MOV     (R4)+,R5        ;POINT TO TABLE
002202  060105                        ADD     R1,R5           ;POINT TO TABLE ENTRY FOR THIS USER
002204  041415                        BIC     @R4,@R5         ;RESET THE DESIRED FLAG
002206  000167  0000000G              JMP     RDCMD
                                        ;
                                        ;  Routine to store a value into a word cell
                                        ;
002212  012405                STRWRD: MOV     (R4)+,R5        ;POINT TO WORD
002214  011415                        MOV     (R4),(R5)       ;SET VALUE IN WORD
002216  000167  0000000G              JMP     RDCMD
                                        ;
                                        ;  Routine to accrue a byte value and store it into a line table.
                                        ;
002222  122327  000075          SETBYT: CMPB    (R3)+,#'=       ;EQUAL SIGN SHOULD BE NEXT CHAR
002226  001014                        BNE     NOEQL           ;BR IF ERROR
002230  122327  000040          2$:     CMPB    (R3)+,#' '      ;SKIP INTERVENING SPACES
002234  001775                        BEQ     2$
002236  005303                        DEC     R3
002240  004767  0000000G              CALL    ACRDEC          ;ACCRUE THE DECIMAL VALUE
002244  116702  0000000G              MOVB    CORUSR,R2       ;GET LINE INDEX #
002250  061402                        ADD     @R4,R2          ;POINT TO LINE ENTRY IN TABLE
002252  110112                        MOVB    R1,@R2          ;STORE VALUE INTO TABLE
002254  000167  0000000G              JMP     RDCMD
                                        ;
002260                          NOEQL:  FABORT  #MISSEQ         ;MISSING EQUAL SIGN
                                        ;
                                        ;  Set EDIT EDIT
                                        ;
002270  042761  000000C 0000000G SEEDIT: BIC    #$TECO!$KED,LSW5(R1)
002276  000167  0000000G              JMP     RDCMD
                                        ;
                                        ;  Set EDIT TECO
                                        ;
002302  042761  0000000G 0000000G SETECO: BIC    #$KED,LSW5(R1)
002310  052761  0000000G 0000000G        BIS    #$TECO,LSW5(R1)
002316  000167  0000000G              JMP     RDCMD
                                        ;
                                        ;  SET EDIT KED/K52
                                        ;
002322                          SEK52:
002322  042761  0000000G 0000000G SEKED:  BIC    #$TECO,LSW5(R1)
002330  052761  0000000G 0000000G        BIS    #$KED,LSW5(R1)
002336  000167  0000000G              JMP     RDCMD
```

```
                                        ;
                                        ;   SET UCL FIRST
                                        ;
002342  052761  000000G 000000G STUCLF: BIS     #$UCLCF,LSW7(R1);Set UCL-first flag
002350  042761  000000C 000000G         BIC     #<$UCLCM!$UCLCL>,LSW7(R1) ;Clear call-middle & call-last flags
002356  000167  000000G                 JMP     RDCMD
                                        ;
                                        ;   SET UCL MIDDLE
                                        ;
002362  052761  000000G 000000G STUCLM: BIS     #$UCLCM,LSW7(R1);Set UCL-middle flag
002370  042761  000000C 000000G         BIC     #<$UCLCF!$UCLCL>,LSW7(R1) ;Clear call-first & call-last flags
002376  000167  000000G                 JMP     RDCMD
                                        ;
                                        ;   SET UCL LAST
                                        ;
002402  052761  000000G 000000G STUCLL: BIS     #$UCLCL,LSW7(R1);Set UCL-last flag
002410  042761  000000C 000000G         BIC     #<$UCLCF!$UCLCM>,LSW7(R1) ;Clear call-first & call-middle flag
002416  000167  000000G                 JMP     RDCMD
                                        ;
                                        ;   SET UCL NONE
                                        ;
002422  042761  000000C 000000G STUCLN: BIC     #<$UCLCF!$UCLCM!$UCLCL>,LSW7(R1) ;Clear all UCL-call flags
002430  000167  000000G                 JMP     RDCMD
                                        ;
                                        ;   SET KMON IND
                                        ;
002434  005767  000000G         SETIND: TST     INDSAV          ;IS IND AVAILABLE?
002440  001432                         BEQ     1$              ;BR IF NOT
002442  052761  000000G 000000G         BIS     #$INDDF,LSW5(R1);SAY WE SHOULD USE IND
002450                                  .GVAL   #XAREA,#<CFSTS-MONVEC>  ;GET CURRENT COMMAND FILE FLAGS
002470  010002                         MOV     R0,R2
002472  052702  000000G                 BIS     #CF$IND,R2              ;SET IND FLAG
002476                                  .PVAL   #XAREA,#<CFSTS-MONVEC>,R2;STORE UPDATED FLAGS
002522  000167  000000G                 JMP     RDCMD
002526                          1$:     FABORT  #NOIND
                                        ;
                                        ;   SET KMON NO IND
                                        ;
002536  042761  000000G 000000G RSTIND: BIC     #$INDDF,LSW5(R1);CLEAR IND FLAG
002544                                  .GVAL   #XAREA,#<CFSTS-MONVEC>  ;GET CURRENT COMMAND FILE FLAGS
002564  010002                         MOV     R0,R2
002566  042702  000000G                 BIC     #CF$IND,R2              ;RESET IND FLAG
002572                                  .PVAL   #XAREA,#<CFSTS-MONVEC>,R2;STORE UPDATED FLAGS
002616  000167  000000G                 JMP     RDCMD
```

```
                        .SBTTL  .       Prompt
                ;-----------------------------------------------------------------------
                ;  SET PROMPT string
                ;
002622          SETPRT:
                ;
                ;  Skip any leading spaces in front of prompt string
                ;
002622  004767  000000G         CALL    SKPSPC
002626  012704  000000G         MOV     #KMPRMT,R4      ;Point to prompt holder in context block
                ;
                ;  Determine if prompt string is enclosed in quotes
                ;
002632  111300                  MOVB    (R3),R0         ;Get first char of prompt string
002634  120027  000047          CMPB    R0,#47          ;Is string enclosed in " ' " characters?
002640  001403                  BEQ     2$              ;Br if yes
002642  120027  000042          CMPB    R0,#42          ;How about " " " ?
002646  001007                  BNE     3$              ;Br if not
                ;
                ;  String is enclosed in quotes
                ;
002650  004767  000000G 2$:     CALL    ACRSTR          ;Accrue the string
002654  010002                  MOV     R0,R2           ;Get length of the string
002656  001417                  BEQ     6$              ;Br if string is empty
002660  012703  000000G         MOV     #BLK0,R3        ;Point to buffer with string
002664  000403                  BR      10$
                ;
                ;  String is not enclosed in quotes
                ;
002666  020302  3$:     CMP     R3,R2           ;Anything in prompt string?
002670  103012                  BHIS    6$              ;Br if not
002672  160302                  SUB     R3,R2           ;Determine length of prompt string
                ;
                ;  Make sure string is not too long
                ;
002674  020227  000000G 10$:    CMP     R2,#MXPRMT      ;Is string too long?
002700  101411                  BLOS    5$              ;Br if ok
002702                          FERR    #BADPMT         ;Prompt string too long
002716  112724  000056 6$:     MOVB    #'.,(R4)+       ;Set prompt to "."
002722  000402                  BR      8$
                ;
                ;  Set prompt string
                ;
002724  112324  5$:     MOVB    (R3)+,(R4)+     ;Move prompt string to KMON context block
002726  077202                  SOB     R2,5$
002730  112714  000200 8$:     MOVB    #200,(R4)       ;Terminate the prompt string
                ;
                ;  Finished
                ;
002734  000167  000000G 9$:     JMP     RDCMD           ;Go get next command
```

```
                                .SBTTL  .       Syspassword
                        ;-----------------------------------------------------------------------
                        ;  SET SYSPASSWORD command
                        ;
002740                  SETSYP:
                        ;
                        ;  User must have SYSPRV privilege
                        ;
002740  004767  000000G         CALL    CKSYPV          ;Make sure user has SYSPRV privilege
                        ;
                        ;  Accrue the password string
                        ;
002744  004767  000000G         CALL    ACRTXT          ;Accrue the text string
                        ;
                        ;  Make sure it's not too long
                        ;
002750  020027  000024          CMP     R0,#20.         ;Compare with max legal length
002754  101404                  BLOS    1$              ;Br if ok
002756                          FABORT  #EM$SPL         ;Password string is too long
                        ;
                        ;  Move password to password buffer
                        ;
002766  012702  000000G 1$:     MOV     #BLK0,R2        ;Point to accrued password
002772  012703  000000G         MOV     #SYPSWD,R3      ;Point to system buffer
002776  112223          2$:     MOVB    (R2)+,(R3)+     ;Move password string
003000  001376                  BNE     2$              ;Loop if more to move
                        ;
                        ;  Finished
                        ;
003002  000167  000000G         JMP     RDCMD
```

```
                                        .SBTTL  .       Endstartup
                                ;-------------------------------------------------------------------------
                                ;  SET ENDSTARTUP command
                                ;  (Signal end of privileged portion of start-up command file)
                                ;
003006  042761  000000G 000000G SETESU: BIC     #$NOIN,LSW3(R1) ;Allow input to be accepted for line
003014  042761  000000G 000000G        BIC     #$SUCF,LSW9(R1) ;Say not in start-up command file
003022  000167  000000G                 JMP     RDCMD
```

```
                                        .SBTTL  .       Window
                                ;-------------------------------------------------------------------
                                ;  SET WINDOW command
                                ;
003026                          SETWIN:
                                ;
                                ;  User must be authorized to create global regions in order to use windows
                                ;
003026  032767  000000G 000000G         BIT     #P2$CGR,PRIVC2  ;May job create global regions?
003034  001004                          BNE     10$             ;Br if yes
003036                                  FABORT  #EM$WCM         ;Not privileged to do this
                                ;
                                ;  Initialize some cells
                                ;
003046  112767  000120  175016  10$:     MOVB    #80.,WINMAK+4   ;Set window width to 80 columns
003054  112767  000020  175011          MOVB    #16.,WINMAK+5   ;Set default max scroll lines
003062  005067  174730                  CLR     WINFLG          ;No flags yet
                                ;
                                ;  Do command parsing
                                ;
003066  012704  001460'                 MOV     #WINHD,R4       ;Point to option driver list
003072  004767  000000G                 CALL    SCNOPS          ;Process all command options
                                ;
                                ;  Now see which options were specified
                                ;
003076  016705  174714          12$:    MOV     WINFLG,R5       ;Pick up option flags
                                ;
                                ;  See if OFF was specified
                                ;
003102  032705  000001                  BIT     #WFDEL,R5       ;Was OFF option specified?
003106  001404                          BEQ     1$              ;Br if not
003110  012700  000104'                 MOV     #WINDEL,R0      ;Delete the window
003114  104375                          EMT     375
003116  000455                          BR      20$
                                ;
                                ;  If WIDE option was specified, set window width to 132 columns
                                ;
003120  032705  000010          1$:     BIT     #WF132,R5       ;Was WIDE option specified?
003124  001403                          BEQ     2$              ;Br if not
003126  112767  000204  174736          MOVB    #132.,WINMAK+4  ;Set window width to 132 columns
                                ;
                                ;  Try to make the window
                                ;
003134  032761  000000C 000000G 2$:     BIT     #<VT100!VT200!VT52>,LTRMTP(R1) ;VT100, VT200 or VT52
003142  001004                          BNE     13$             ;Br if yes
003144                                  FABORT  #EM$SLT         ;Invalid terminal type
003154  012700  000066'          13$:    MOV     #WINMAK,R0      ;Point to window-make arg block
003160  104375                          EMT     375             ;Try to create the window
003162  103435                          BCS     30$             ;Br if error on window creation
                                ;
                                ;  Select the window
                                ;
003164  012700  000100'                 MOV     #WINMAP,R0      ;EMT to map to the window
003170  104375                          EMT     375
                                ;
                                ;  If WIDE was specified, send string to put term in 132 col mode
                                ;
```

```
     58 003172  032705  000010                  BIT     #WF132,R5       ;Was WIDE option specified?
     59 003176  001403                          BEQ     3$              ;Br if not
     60 003200                                  .PRINT  #TC132          ;Put terminal in 132 column mode
     61                                 ;
     62                                 ;  If NARROW was specified, send string to put term in 80 col mode
     63                                 ;
     64 003206  032705  000020          3$:     BIT     #WF80,R5        ;Was NARROW option specified?
     65 003212  001403                          BEQ     4$              ;Br if not
     66 003214                                  .PRINT  #TC80           ;Put terminal in 80 column mode
     67                                 ;
     68                                 ;  If REVERSE was specified, put terminal in reverse video mode
     69                                 ;
     70 003222  032705  000002          4$:     BIT     #WFREV,R5       ;Was REVERSE option specified?
     71 003226  001403                          BEQ     5$              ;Br if not
     72 003230                                  .PRINT  #TCREV          ;Put terminal in reverse video mode
     73                                 ;
     74                                 ;  If NORMAL was specified, put terminal in normal video mode
     75                                 ;
     76 003236  032705  000004          5$:     BIT     #WFNORM,R5      ;Was NORMAL option specified?
     77 003242  001403                          BEQ     20$             ;Br if not
     78 003244                                  .PRINT  #TCNORM         ;Put terminal in normal video mode
     79                                 ;
     80                                 ;  Finished
     81                                 ;
     82 003252  000167  000000G         20$:    JMP     RDCMD
     83                                 ;
     84                                 ;  Error occurred which making window
     85                                 ;
     86 003256  113702  000000G         30$:    MOVB    @#ERRLOC,R2     ;Get EMT error code
     87 003262  020227  000004                  CMP     R2,#MAXWEM      ;Is it too large?
     88 003266  103401                          BLO     31$             ;Br if not
     89 003270  005002                          CLR     R2              ;Use message for error 0 if too big
     90 003272  006302                  31$:    ASL     R2              ;Get word table index
     91 003274  016202  003306'                 MOV     WEM(R2),R2      ;Get address of error message
     92 003300                                  FABORT  R2              ;Print the error message
     93                                 ;
     94                                 ;  Table of window-creation error messages
     95                                 ;
     96 003306  000000G                 WEM:    .WORD   EM$WC0
     97 003310  000000G                         .WORD   EM$WC1
     98 003312  000000G                         .WORD   EM$WC2
     99 003314  000000G                         .WORD   EM$WC3
    100        000004                  MAXWEM  =       <.-WEM>/2
```

```
                                ;-------------------------------------------------------------------------------
                                ;  Process COLUMN=n option of SET WINDOW command
                                ;
003316  010146                  WINCOL: MOV     R1,-(SP)
                                ;
                                ;  Accrue the column parameter value
                                ;
003320  004767  000000G               CALL    ACRDEC
                                ;
                                ;  Store into EMT argument block
                                ;
003324  110167  174542                MOVB    R1,WINMAK+4     ;Set column width in EMT arg block
                                ;
                                ;  Finished
                                ;
003330  012601                        MOV     (SP)+,R1
003332  000207                        RETURN

                                ;-------------------------------------------------------------------------------
                                ;  Process SCROLL[=n] option which specifies the maximum number
                                ;  of scroll lines.
                                ;
003334  010146                  WINSCR: MOV     R1,-(SP)
                                ;
                                ;  If no parameter is specified, allow unlimited scrolling
                                ;
003336  004767  000000G               CALL    SKPSPC          ;Skip over spaces
003342  121327  000075                CMPB    (R3),#'=        ;Value specified?
003346  001005                        BNE     3$              ;If not, allow unlimited
                                ;
                                ;  Accrue the parameter value
                                ;
003350  004767  000000G         1$:   CALL    ACRDEC          ;Accrue decimal parameter
003354  020127  000177                CMP     R1,#127.        ;Asked for more than max?
003360  101402                        BLOS    2$              ;If not, set the requested number
003362  012701  177777          3$:   MOV     #-1,R1          ;Allow unlimited scrolling
                                ;
                                ;  Store value into EMT argument block
                                ;
003366  110167  174501          2$:   MOVB    R1,WINMAK+5     ;Set max scroll lines
                                ;
                                ;  Finished
                                ;
003372  012601                        MOV     (SP)+,R1
003374  000207                        RETURN

                                ;-------------------------------------------------------------------------------
                                ;  SET WINDOW/NOSCROLL
                                ;
003376  105067  174471          WINNOS: CLRB    WINMAK+5        ;0 scroll lines ==> No scrolling
003402  000207                        RETURN

                                ;-------------------------------------------------------------------------------
                                ;  Process a SET WINDOW option that just sets a flag bit
                                ;
003404  051467  174406          WINBIT: BIS     (R4),WINFLG     ;Set appropriate flag bit
003410  000207                        RETURN
```

```
                                        .SBTTL  .       Printwindow
                                ;-------------------------------------------------------------------------------
                                ;  SET PRINTWINDOW/DEVICE=device/TYPE=type
                                ;
003412                          SETPRS:
                                ;
                                ;  Process qualifiers specified with command
                                ;
003412  012704  001574'                MOV     #PRSHD,R4       ;Point to option driver list
003416  004767  0000000G                CALL    SCNOPS          ;Process all of the command options
                                ;
                                ;  Finished
                                ;
003422  000167  0000000G                JMP     RDCMD
                                ;
                                ;  Process the /DEVICE=device qualifer
                                ;
003426  010246                  PRSDEV: MOV     R2,-(SP)
003430  052767  0000000G 0000000G       BIS     #PA$FLG,JPWFLG  ;Default to using flag pages on new device
003436  004767  0000000G              CALL    CHKEQ           ;Equal sign should follow qualifier name
003442  004767  0000000G              CALL    GTRD50          ;Accrue the device name
003446  122327  000072                CMPB    (R3)+,#':       ;Colon specified with device name?
003452  001401                        BEQ     1$              ;Br if yes
003454  005303                        DEC     R3              ;Backup pointer
003456  016767  0000000G 0000000G 1$:  MOV     R50BUF,JPWDEV   ;Store into job context cell
003464  012602                        MOV     (SP)+,R2
003466  000207                        RETURN
                                ;
                                ;  Process the /TYPE=type qualifier
                                ;
003470  010446                  PRSTYP: MOV     R4,-(SP)
003472  004767  0000000G              CALL    CHKEQ           ;Make sure we have an equal sign
003476  012704  001670'                MOV     #PWTHD,R4       ;Point to table of printer types
003502  004767  0000000G              CALL    SEARCH          ;Accrue and look up printer name
003506  103406                        BCS     1$              ;Br if invalid printer type
003510  111467  0000000G              MOVB    (R4),JPWTYP     ;Set printer type code
003514  016467  000002  0000000G       MOV     2(R4),JPWFLG    ;Set printer attribute flags
003522  000417                        BR      9$
003524  005704                  1$:   TST     R4              ;Printer unrecognized or ambigious?
003526  001407                        BEQ     2$              ;Br if not recognized
003530                                FERR    #EM$PTA         ;Printer type ambigious
003544  000406                        BR      9$
003546                          2$:   FERR    #EM$PTU         ;Printer type unrecognized
                                ;
                                ;  Finished
                                ;
003562  012604                  9$:   MOV     (SP)+,R4
003564  000207                        RETURN
                                ;
                                ;  Process the /LETTERQUALITY qualifier
                                ;
003566  052767  0000000G 0000000G PRSLET: BIS     #PA$LET,JPWFLG  ;Set letter-quality flag
003574  000207                        RETURN
                                ;
                                ;  Process the /DRAFT qualifier
                                ;
003576  042767  0000000G 0000000G PRSDRF: BIC     #PA$LET,JPWFLG  ;Clear letter-quality flag
```

```
     58 003604  000207                              RETURN
     59                                 ;
     60                                 ;  Process the /BELL qualifier
     61                                 ;
     62 003606  052767  000000G 000000G PRSBEL:  BIS     #PA$BEL,JPWFLG  ;Set bell flag
     63 003614  000207                              RETURN
     64                                 ;
     65                                 ;  Process the /NOBELL qualifier
     66                                 ;
     67 003616  042767  000000G 000000G PRSNBL:  BIC     #PA$BEL,JPWFLG  ;Clear bell flag
     68 003624  000207                              RETURN
     69                                 ;
     70                                 ;  Process the /FLAG qualifier
     71                                 ;
     72 003626  052767  000000G 000000G PRSFLG:  BIS     #PA$FLG,JPWFLG  ;Set flagpage flag
     73 003634  000207                              RETURN
     74                                 ;
     75                                 ;  Process the /NOFLAG qualifier
     76                                 ;
     77 003636  042767  000000G 000000G PRSNFL:  BIC     #PA$FLG,JPWFLG  ;Clear flagpage flag
     78 003644  000207                              RETURN
     79                                 ;
     80                                 ;  Process the /STAMP qualifier
     81                                 ;
     82 003646  052767  000000G 000000G PRSSTM:  BIS     #PA$DTS,JPWFLG  ;Set date/time stamp flag
     83 003654  000207                              RETURN
     84                                 ;
     85                                 ;  Process the /NOSTAMP qualifier
     86                                 ;
     87 003656  042767  000000G 000000G PRSNST:  BIC     #PA$DTS,JPWFLG  ;Clear date/time stamp flag
     88 003664  000207                              RETURN
     89                                 ;
     90                                 ;  Process the /WIDTH qualifier
     91                                 ;
     92 003666  042767  000000G 000000G PRSWID:  BIC     #PA$NWD,JPWFLG  ;Clear nowidth flag
     93 003674  000207                              RETURN
     94                                 ;
     95                                 ;  Process the /NOWIDTH qualifier
     96                                 ;
     97 003676  052767  000000G 000000G PRSNWD:  BIS     #PA$NWD,JPWFLG  ;Set nowidth flag
     98 003704  000207                              RETURN
     99                                 ;
    100                                 ;  Process the /KEYPRINT qualifier
    101                                 ;
    102 003706  116701  000000G         PRSKEY:  MOVB    CORUSR,R1       ;Get job index number
    103 003712  052761  000000G 000000G         BIS     #$PWKEY,LSW11(R1) ;Set KEYPRINT flag
    104 003720  000207                           RETURN
    105                                 ;
    106                                 ;  Process the /NOKEYPRINT qualifier
    107                                 ;
    108 003722  116701  000000G         PRSNKY:  MOVB    CORUSR,R1       ;Get job index number
    109 003726  042761  000000G 000000G         BIC     #$PWKEY,LSW11(R1) ;Clear KEYPRINT flag
    110 003734  000207                           RETURN
```

```
                                        .SBTTL  .       Priority
                                ;-------------------------------------------------------------------------
                                ;  SET PRIORITY value
                                ;
003736  004767  000000G         SETPRI: CALL    ACRDEC          ;Accrue decimal value
003742  020127  000000G                 CMP     R1,#MAXPRI      ;Is priority too large?
003746  101425                         BLOS    3$              ;Br if ok
003750                          5$:     FERR    #BADPRI         ;Invalid priority value
003764  012705  000000                 MOV     #0,R5           ;Print minimum priority
003770  004767  000000G                CALL    PRTDEC
003774                                  .PRINT  #TOTXT          ;Print " to "
004002  116705  000000G                MOVB    MXJPRI,R5       ;Print maximum priority
004006  004767  000000G                CALL    PRTDEC
004012                                  .PRINT  #CRLF           ;End print line
004020  000427                         BR      9$
004022  120167  000000G         3$:     CMPB    R1,MXJPRI       ;Does it exceed max allowed for job?
004026  101417                         BLOS    2$              ;Br if ok
004030                                  FERR    #HIPRI          ;Priority is too high
004044  116705  000000G                MOVB    MXJPRI,R5       ;Get max allowed value
004050  004767  000000G                CALL    PRTDEC          ;Display max priority
004054                                  .PRINT  #CRLF
004062  116701  000000G                MOVB    MXJPRI,R1       ;Get max allowed for job
004066  012700  000062'         2$:     MOV     #PRIEMT,R0      ;Point to EMT arg block to set job priority
004072  010160  000002                 MOV     R1,2(R0)        ;Store prio value into EMT arg block
004076  104375                         EMT     375             ;Set job priority
004100  000167  000000G         9$:     JMP     RDCMD           ;Go get next command
```

```
                                        .SBTTL  .       Maxpriority
                                ;-------------------------------------------------------------------------
                                ;  SET MAXPRIORITY value
                                ;
004104  004767  000000G         SETMPR: CALL    ACRDEC          ;Accrue decimal value
004110  020127  000000G                 CMP     R1,#MAXPRI      ;Larger than max allowed?
004114  101425                          BLOS    1$              ;Br if ok
004116                                  FERR    #BADPRI         ;Invalid priority value
004132  005000                          CLR     R0              ;Get minimum value
004134  004767  000000G                 CALL    PRTDEC          ;Print minimum value
004140                                  .PRINT  #TOTXT          ;Print " to "
004146  012705  000000G                 MOV     #MAXPRI,R5      ;Get maximum value
004152  004767  000000G                 CALL    PRTDEC          ;Print it
004156                                  .PRINT  #CRLF
004164  116701  000000G                 MOVB    VPRIDF,R1       ;Set to default value
004170  120167  000000G         1$:     CMPB    R1,MXJPRI       ;Is he attempting to increase the max pri?
004174  101404                          BLOS    2$              ;Br if not
004176                                  FABORT  #EM$CIP         ;Cannot increase maxpri
004206  110167  000000G         2$:     MOVB    R1,MXJPRI       ;Set maximum job priority
004212  116702  000000G                 MOVB    CORUSR,R2       ;Get current job index number
004216  126201  000000G                 CMPB    LBSPRI(R2),R1   ;Is current priority too high?
004222  101405                          BLOS    3$              ;Br if ok
004224  012700  000062'                 MOV     #PRIEMT,R0      ;Point to EMT arg block to set prio
004230  010160  000002                  MOV     R1,2(R0)        ;Set new priority for job
004234  104375                          EMT     375
004236  000167  000000G         3$:     JMP     RDCMD           ;Finished
```

```
                                        .SBTTL  .       Error
                                ;-------------------------------------------------------------------------------
                                ;  ROUTINE TO SET ERROR SEVERITY ABORT LEVEL
                                ;
004242  111467  0000006         SETSEV: MOVB    @R4,ERRSEV      ;SET ERROR SEVERITY FOR USER
004246  111404                          MOVB    @R4,R4          ;GET ERROR SEVERITY LEVEL
004250  042704  177400                  BIC     #177400,R4      ;CLEAR HIGH ORDER BYTE
004254  052704  001400                  BIS     #3*400,R4       ;SET COMMAND FILE NESTING DEPTH
004260                                  .PVAL   #XAREA,#<CCFABLV-MONVEC>,R4 ;SET ABORT LEVEL IN SIMUL RMON
004304  000167  0000006                 JMP     RDCMD

                                        .SBTTL  .       Shutdown
                                ;-------------------------------------------------------------------------------
                                ;  Set the flag that says a system shutdown is being done.
                                ;
004310  004767  0000006         SETSHT: CALL    CKPRIV          ;User must have privilege
004314  112767  177777  0000006         MOVB    #-1,STPFLG      ;Say system shutdown taking place
004322  005067  0000006                 CLR     BOTDEV          ;Reboot from system disk
004326  000167  0000006                 JMP     RDCMD

                                ;-------------------------------------------------------------------------------
                                ;  Clear the flag that says a shutdown is taking place.
                                ;
004332  004767  0000006         SETNSH: CALL    CKPRIV          ;User must have privilege
004336  105067  0000006                 CLRB    STPFLG          ;Say we are not doing a shutdown
004342  000167  0000006                 JMP     RDCMD
```

```
                                        .SBTTL  .       Log
                                ;-------------------------------------------------------------------------
                                ;  Process the SET LOG command.
                                ;
004346                          SETLOG:
                                ;
                                ;  Accrue option keyword
                                ;
004346  012704  002350'                 MOV     #STLGHD,R4      ;Point to keyword table
004352  004767  0000000G                CALL    SCNOPS          ;Process command qualifiers
004356  000167  0000000G                JMP     RDCMD           ;Finished with command
                                ;
                                ;  SET LOG CLOSE
                                ;
004362  004767  0000000G        STLGCL: CALL    LOGCLS          ;Close the log file
004366  000207                          RETURN
                                ;
                                ;  SET LOG CLEAN
                                ;
004370  010346                  STLGCN: MOV     R3,-(SP)
004372  032767  0000000G 0000000G        BIT     #LF$OPN,LOGFLG  ;Is the log file open now?
004400  001427                          BEQ     9$              ;Br if not
004402  012767  0000000G 0000000G        MOV     #LOGBUF,LOGPTR  ;Reset buffer pointer
004410  005067  0000000G                CLR     LOGBLK          ;Start writing at block 0
004414  012703  0000000G                MOV     #LOGCHN,R3      ;Get channel # used for log file
004420  016700  0000000G                MOV     CXTRMN,R0       ;Point to simulated RMON area
004424  062700  0000000G                ADD     #R$CHN,R0       ;Point to channel block for channel 0
004430  020327  000021                  CMP     R3,#17.         ;Is this channel in extended chan area?
004434  103404                          BLO     1$              ;Br if not
004436  162703  000021                  SUB     #17.,R3         ;Get channel # relative to extended channels
004442  062700  0000000C                ADD     #R$XCHN-R$CHN,R0;Point to 1st extended channel block
004446  070327  0000000G        1$:     MUL     #CHNSIZ,R3      ;Get offset to block of interest
004452  060003                          ADD     R0,R3           ;Point to our channel block
004454  005063  0000000G                CLR     C.USED(R3)      ;Say no data has been written to file
004460  012603                  9$:     MOV     (SP)+,R3
004462  000207                          RETURN
                                ;
                                ;  SET LOG WRITE
                                ;
004464  052767  0000000G 0000000G STLGWR: BIS    #LF$WRT,LOGFLG  ;Enable writes to log file
004472  000207                          RETURN
                                ;
                                ;  SET LOG NOWRITE
                                ;
004474  042767  0000000G 0000000G STLGNW: BIC    #LF$WRT,LOGFLG  ;Disable log file
004502  000207                          RETURN
                                ;
                                ;  SET LOG INPUT
                                ;
004504  042767  0000000G 0000000G STLGIN: BIC    #LF$OUT,LOGFLG  ;Clear output-logging flag
004512  052767  0000000G 0000000G        BIS     #LF$IN,LOGFLG   ;Set input-logging flag
004520  000207                          RETURN
                                ;
                                ;  SET LOG OUTPUT
                                ;
004522  042767  0000000G 0000000G STLGOT: BIC    #LF$IN,LOGFLG   ;Clear input-logging flag
004530  052767  0000000G 0000000G        BIS     #LF$OUT,LOGFLG  ;Set output-logging flag
```

```
      58 004536  000207                                RETURN
      59                                ;
      60                                ;  SET LOG ALL
      61                                ;
      62 004540  052767  000000C 000000G STLGAL:  BIS     #LF$IN!LF$OUT,LOGFLG ;Log both input and output
      63 004546  000207                           RETURN
```

```
                                        ;
                                        ;  SET LOG FILE=file-spec
                                        ;
004550  010446                  STLGFL: MOV     R4,-(SP)
004552  010546                          MOV     R5,-(SP)
004554  004767  000000G                 CALL    LOGCLS          ;Close current log file
004560  004767  000000G                 CALL    SKPSPC          ;Skip over spaces
004564  122327  000075                  CMPB    (R3)+,#'=       ;See if equal sign was specified
004570  001401                          BEQ     1$              ;Br if yes (skip over it)
004572  005303                          DEC     R3              ;Point back to first char of file spec
                                        ;
                                        ;  Accrue the file spec
                                        ;
004574  012704  000000G           1$:    MOV     #R50LOG,R4      ;Point to default extension ("LOG")
004600  012705  000001                  MOV     #1,R5           ;Tell ACRFIL this is an output file
004604  004767  000000G                 CALL    ACRFIL          ;Accrue the file spec
004610  103514                          BCS     10$             ;Br if invalid file spec
                                        ;
                                        ;  Translate logical device name to physical
                                        ;
004612  012705  000000G                 MOV     #FILNAM,R5      ;Point to file name buffer
004616  004767  000000G                 CALL    LOGASN          ;Perform logical name translation
                                        ;
                                        ;  Check for empty file name
                                        ;
004622  012767  000000G 000222          MOV     #DS$DIR,DSBLOK+0 ;Assume device is file struct
                                ;This forces it to check for file name if device is unrecognized by .DSTAT
004630                                  .DSTAT  #DSBLOK,#FILNAM ;Find out what kind of device it is
004642  032767  000000G 000202          BIT     #DS$DIR,DSBLOK+0 ;Standard file structured device?
004650  001403                          BEQ     5$              ;If not, can be opened without file name
004652  005767  000002G                 TST     FILNAM+2        ;Was a file name specified?
004656  001471                          BEQ     10$             ;Error if not file name on dir struct device
                                        ;
                                        ;  Don't allow log output to go to "TT:"
                                        ;
004660  016700  000000G           5$:    MOV     FILNAM,R0       ;Get the log device name
004664  004767  000000G                 CALL    CHKTTD          ;Is the log device TT?
004670  103464                          BCS     10$             ;Br if yes -- Error
                                        ;
                                        ;  Try to open the file
                                        ;
004672                                  .ENTER  #XAREA,#LOGCHN,#FILNAM,FILNAM+8.
004720  103004                          BCC     2$              ;Br if enter ok
004722                                  FABORT  #BDLGOP         ;Cannot open log file
                                        ;
                                        ;  Save information about the log file physical device
                                        ;
004732  016705  000000G           2$:    MOV     FILNAM,R5       ;Get log file device name
004736  004767  000000G                 CALL    CHKDEV          ;Convert to dev # and unit #
004742  103421                          BCS     3$              ;Br if invalid device
004744  020467  000000G                 CMP     R4,LDDEVX       ;Is log file on a logical disk?
004750  001407                          BEQ     4$              ;Br if yes
004752  110467  000000G                 MOVB    R4,LOGDVU       ;Save log file device #
004756  110067  000001G                 MOVB    R0,LOGDVU+1     ;Save log file unit #
004762  005067  000000G                 CLR     LOGBAS          ;Say not on a logical disk
004766  000407                          BR      3$
004770  006300                  4$:     ASL     R0              ;Convert unit # to word table index
```

```
     58 004772  016067  000000G 000000G          MOV     LDPDEV(R0),LOGDVU ;Save physical device and unit #
     59 005000  016067  000000G 000000G          MOV     LDBASE(R0),LOGBAS ;Save base block # of logical disk
     60                                     ;
     61                                     ;  Set "hold" mode for spooled log files
     62                                     ;
     63 005006  012700  000000G             3$:  MOV     #SPLHLA,R0        ;Point to emt argument block
     64 005012  104375                           EMT     375               ;Set hold mode for log file
     65                                     ;
     66                                     ;  Initialize the log file
     67                                     ;
     68 005014  052767  000000C 000000G          BIS     #<LF$OPN!LF$WRT!LF$IN!LF$OUT>,LOGFLG ;Say log file is open
     69 005022  012767  000000G 000000G          MOV     #LOGBUF,LOGPTR    ;Init pointer to log buffer
     70 005030  005067  000000G                  CLR     LOGBLK            ;Write to block 0
     71                                     ;
     72                                     ;  Finished
     73                                     ;
     74 005034  012605                           MOV     (SP)+,R5
     75 005036  012604                           MOV     (SP)+,R4
     76 005040  000207                           RETURN
     77                                     ;
     78                                     ;  Error -- Invalid file spec
     79                                     ;
     80 005042                              10$: FABORT  #BDFNAM           ;Invalid file name
     81 005052  000000  000000  000000   DSBLOK: .WORD   0,0,0,0,0          ;.DSTAT result block
        005060  000000  000000
```

```
                                        .SBTTL  .       Logoff
                                ;-----------------------------------------------------------------------
                                ;  Process the SET LOGOFF command.
                                ;  This command is used to declare a logoff command file.
                                ;
005064                          SETLOF:
                                ;
                                ;  Accrue option keyword
                                ;
005064  012704  002414'                 MOV     #STLOHD,R4      ;Point to keyword table
005070  004767  0000000G                CALL    SEARCH          ;Identify keyword
005074  103002                         BCC     1$              ;Br if identified keyword
005076  000167  173352                 JMP     BDSO            ;Invalid option
                                ;
                                ;  Enter keyword processing routine
                                ;
005102  000134                  1$:    JMP     @(R4)+          ;Enter keyword processing routine
                                ;
                                ;  SET LOGOFF FILE=file-spec
                                ;
005104  032761  0000000G 0000000G STLOFL: BIT    #$SUCF,LSW9(R1) ;Are we in a startup command file?
005112  001010                         BNE     1$              ;Br if yes
005114  032767  0000000G 0000000G       BIT     #PO$SYS,PRIVCO  ;Do we have SYSPRV privilege?
005122  001004                         BNE     1$              ;Br if yes
005124                                  FABORT  #EM$NSF         ;Not in a startup command file
                                ;
                                ;  Accrue the file spec
                                ;
005134  004767  0000000G        1$:    CALL    SKPSPC          ;Skip over spaces
005140  122327  000075                CMPB    (R3)+,#'=       ;Should have equal sign
005144  001401                         BEQ     3$              ;Br if yes (skip over it)
005146  005303                         DEC     R3              ;Point to 1st char of file name
005150  012704  0000000G        3$:    MOV     #R50COM,R4      ;Set "COM" as default extension
005154  005005                         CLR     R5              ;Tell ACRFIL this is an input file
005156  004767  0000000G               CALL    ACRFIL          ;Accrue the file spec
005162  103416                         BCS     10$             ;Br if invalid file spec
                                ;
                                ;  Translate logical file device name to physical device
                                ;
005164  012705  0000000G               MOV     #FILNAM,R5      ;Point to file spec
005170  004767  0000000G               CALL    LOGASN          ;Perform any logical device assignment
                                ;
                                ;  Move file spec to job context block
                                ;
005174  012705  0000000G               MOV     #FILNAM,R5      ;Point to file spec
005200  012704  0000000G               MOV     #LOFSPC,R4      ;Point to job context area
005204  012700  000004                 MOV     #4,R0           ;Get # words to move
005210  012524                  2$:    MOV     (R5)+,(R4)+     ;Move file spec to job context block
005212  077002                         SOB     R0,2$
                                ;
                                ;  Finished
                                ;
005214  000167  0000000G               JMP     RDCMD
                                ;
                                ;  Invalid file spec
                                ;
005220                          10$:    FABORT  #BDFNAM         ;Invalid file name
```

```
                                        .SBTTL  .       Subprocess
                                ;-----------------------------------------------------------------------
                                ;  Process the SET SUBPROCESS command.
                                ;  This command is used to declare a command file to be executed when
                                ;  a sub process (virtual line) is initiated.
                                ;
005230                          SETSBP:
                                ;
                                ;  Accrue option keyword
                                ;
005230  012704  002424'                 MOV     #SBPHD,R4       ;Point to keyword table
005234  004767  0000000G                CALL    SCNOPS          ;Process the options
005240  000167  0000000G                JMP     RDCMD
                                ;
                                ;  SET SUBPROCESS/FILE=file-spec
                                ;
005244  010446                  SBPFIL: MOV     R4,-(SP)
005246  010546                          MOV     R5,-(SP)
                                ;
                                ;  This command is only legal within a start-up command file
                                ;
005250  032761  0000000G 0000000G       BIT     #$SUCF,LSW9(R1) ;Are we in a startup command file?
005256  001010                          BNE     1$              ;Br if yes
005260  032767  0000000G 0000000G       BIT     #PO$SYS,PRIVCO  ;Do we have SYSPRV privilege?
005266  001004                          BNE     1$              ;Br if yes
005270                                  FABORT  #EM$NSF         ;Not in a startup command file
                                ;
                                ;  Accrue the file spec
                                ;
005300  004767  0000000G        1$:     CALL    ACRTXT          ;Accrue the file string
005304  020027  000017                  CMP     R0,#15.         ;Compare with max legal length
005310  101404                          BLOS    3$              ;Br if ok
005312                                  FABORT  #EM$STL         ;File spec is too long
                                ;
                                ;  Move file spec to job context block
                                ;
005322  012705  0000000G        3$:     MOV     #BLK0,R5        ;Point to file spec
005326  012704  0000000G                MOV     #SBPSUF,R4      ;Point to job context area
005332  112524                  2$:     MOVB    (R5)+,(R4)+     ;Move file spec to job context block
005334  001376                          BNE     2$              ;Loop till all moved
                                ;
                                ;  Finished
                                ;
005336  012605                          MOV     (SP)+,R5
005340  012604                          MOV     (SP)+,R4
005342  000207                          RETURN
                                ;
                                ;  Invalid file spec
                                ;
005344                          10$:    FABORT  #BDFNAM         ;Invalid file name
```

```
                                        .SBTTL  .       Kmon
                                ;-------------------------------------------------------------------------
                                ;  Process the SET KMON USER command.
                                ;  Verify that a user-written command interface program exists and if so,
                                ;  set a flag saying to use it for this job.
                                ;
005354                          SETUKM:
                                ;
                                ;  See if user specified a file spec
                                ;
005354  004767  0000000G              CALL    SKPSPC          ;Skip over any spaces
005360  105713                       TSTB    (R3)            ;Anything specified following "USER"?
005362  001417                       BEQ     2$              ;Br if not
005364  122327  000075               CMPB    (R3)+,#'=       ;Equal sign?
005370  001053                       BNE     3$              ;Error if not
                                ;
                                ;  Accrue the user file name
                                ;
005372  012704  0000000G             MOV     #R50SAV,R4      ;Set default extension
005376  005005                       CLR     R5              ;Tell ACRFIL this is an input file
005400  004767  0000000G             CALL    ACRFIL          ;Accrue the file spec
005404  103451                       BCS     4$              ;Br if invalid file spec
005406  012704  0000000G             MOV     #FILNAM,R4      ;Point to area with file spec
005412  010405                       MOV     R4,R5
005414  004767  0000000G             CALL    LOGASN          ;Convert logical device name to physical
005420  000404                       BR      6$
                                ;
                                ;  User did not specify a file name.  Default to "SY:UKMON.SAV".
                                ;
005422  012704  0000000G     2$:     MOV     #UCIDEF,R4      ;Point to default file spec
005426  016714  0000000G             MOV     SYNAME,(R4)     ;Set physical SY device
                                ;
                                ;  Move file spec to UCISPC area in context block
                                ;
005432  012700  000004       6$:     MOV     #4,R0           ;Get # words to move
005436  012705  0000000G             MOV     #UCISPC,R5      ;Store UCI file spec here
005442  012425               5$:     MOV     (R4)+,(R5)+     ;Move file spec to job context block
005444  077002                       SOB     R0,5$
                                ;
                                ;  Try to lookup the specified file
                                ;
005446                                  .LOOKUP #XAREA,#1,#UCISPC;Try to find command processor program
005466  103410                       BCS     1$              ;Br if not available
005470                                  .CLOSE  #1              ;Close the program file
005476  052761  0000000G 0000000G     BIS     #$UKMON,LSW7(R1);Set flag saying to use user-written program
005504  000167  0000000G             JMP     RDCMD
                                ;
                                ;  Error processing
                                ;  Cannot find specified file
                                ;
005510                          1$:     FABORT  #EM$NUK         ;User-written command processor not available
                                ;
                                ;  Invalid syntax
                                ;
005520                          3$:     FABORT  #MISSEQ
                                ;
                                ;  Invalid file spec
```

```
      58                                ;
      59 005530                         4$:     FABORT  #BDFNAM
```

```
                                ;-------------------------------------------------------------------------------
                                ;  Process the SET KMON NEW command.
                                ;  Begin to use a new version of TSKMON.
                                ;
005540  004767  000000G        SETKNW: CALL    CKSYPV              ;Must have SYSPRV privilege
005544                                  .LOOKUP #XAREA,#1,#KMNNAM ;TRY TO LOOKUP SY:TSKMON.SAV
005564  103500                        BCS     1$                  ;BR IF NOT THERE
                                ;
                                ;  Get information about TSKMON out of block 0 of file
                                ;
005566                                  .READW  #XAREA,#1,#BLK0,#256.,#0 ;READ IN BLOCK 0
005624  016700  0000500G              MOV     BLK0+50,R0          ;GET TOP ADDRESS OF KMON
005630  062700  000003                ADD     #3,R0               ;BOUND UP TO NEXT WORD
005634  042700  000001                BIC     #1,R0               ;FORCE EVEN
005640  010067  000000G               MOV     R0,KMNTOP
005644  162700  000000G               SUB     #KMNBAS,R0          ;BASE ADDRESS OF KMON
005650  010067  000000G               MOV     R0,KMNHI            ;TOP OF TSKMON-KMNBAS
005654  062700  000777                ADD     #511.,R0            ;BOUND UP TO PAGE SIZE
005660  000241                        CLC
005662  006000                        ROR     R0                  ;CVT TO # WORDS
005664  000300                        SWAB    R0                  ;CVT TO # PAGES
005666  042700  177400                BIC     #^C377,R0
005672  062700  000000G               ADD     #CXTPAG,R0          ;# PAGES NEEDED TO JOB CONTEXT AREA
005676  010067  000000G               MOV     R0,KMNPGS           ;# 256 WORD PAGES FOR KMON & CONTEXT
005702  016767  0000420 0000000       MOV     BLK0+42,KMNSTK      ;INITIAL STACK POINTER
005710  016767  0000400 0000000       MOV     BLK0+40,KMNSTR      ;STARTING ADDRESS
                                ;
                                ;  Now do .SAVESTATUS so we can quickly reopen to kmon
                                ;
005716                                  .SAVEST #XAREA,#1,#KMNCHN ;SAVE STATUS
                                ;
                                ;  Now purge and reopen channel 17 to the new file (for overlays).
                                ;
005736                                  .PURGE  #17               ;Purge channel 17 (overlay channel)
005744                                  .REOPEN #XAREA,#17,#KMNCHN ;Open channel 17 to new file
                                ;
                                ;  Exit to reload Kmon
                                ;
005764                                  .EXIT                     ;EXIT AND REENTER NEW VERSION OF TSKMON
                                ;
                                ;  Error: Cannot find new Kmon file
                                ;
005766                          1$:     FABORT  #NOKMON           ;ERROR -- CAN'T FIND TSKMON

                                ;-------------------------------------------------------------------------------
                                ;  Process the SET CCL NEW command.
                                ;  Begin to use a enw version of CCL
                                ;
005776  004767  000000G        SETCNW: CALL    CKSYPV              ;Must have SYSPRV privilege
006002                                  .LOOKUP #XAREA,#1,#CCLNAM ;TRY TO LOOKUP SY:CCL.SAV
006022  103412                        BCS     1$                  ;BR IF CAN'T FIND IT
006024                                  .SAVEST #XAREA,#1,#CCLSAV ;SAVE STATUS
006044  000167  000000G               JMP     RDCMD               ;FINISHED
006050                          1$:     FABORT  #NOCCL            ;CAN'T FIND CCL.SAV
```

```
                                        .SBTTL  .       LD
                                ;-------------------------------------------------------------------------------
                                ;  Process SET LDn command
                                ;
006060  005002                  SETLD:  CLR     R2              ;ASSUME UNIT # = 0
006062  020067  000000G                 CMP     R0,R50LD        ;IS UNIT "LD"?
006066  001404                          BEQ     1$              ;BR IF YES
006070  010002                          MOV     R0,R2           ;GET DEVICE NAME
006072  166702  000000G                 SUB     R50LD0,R2       ;SUBTRACT "LD0" TO GET UNIT # IN R2
006076  006302                          ASL     R2              ;CONVERT TO WORD TABLE INDEX
                                ;
                                ;  Get the option word
                                ;
006100  012704  001264'         1$:     MOV     #LDOPHD,R4      ;POINT TO OPTION TABLE
006104  004767  000000G                 CALL    SEARCH          ;ACCRUE AND CHECK OPTION WORD
006110  103004                          BCC     2$              ;BR IF FOUND
006112                                  FABORT  #CSIMS1         ;INVALID OPTION WORD
                                ;
                                ;  Jump off to option processing routine
                                ;
006122  000134                  2$:     JMP     @(R4)+          ;ENTER PROCESSING ROUTINE
                                ;
                                ;  SET LD CLEAN
                                ;
006124  004767  000000G         SLDCLN: CALL    LDCLEN          ;CLEAN UP LOGICAL DISKS
006130  000167  000000G                 JMP     RDCMD
                                ;
                                ;  SET LDn WRITE
                                ;
006134  042762  000000G 000000G SLDWRT: BIC     #LD$RON,LDFLAG(R2) ;CLEAR READ-ONLY FLAG
006142  000167  000000G                 JMP     RDCMD           ;FINISHED
                                ;
                                ;  SET LDn NOWRITE
                                ;
006146  052762  000000G 000000G SLDNWR: BIS     #LD$RON,LDFLAG(R2) ;SET READ-ONLY FLAG
006154  000167  000000G                 JMP     RDCMD           ;FINISHED
                                ;
                                ;  SET LDn FREE
                                ;
006160  004767  000052          SLDFRE: CALL    LDDMT           ;DISMOUNT THIS LD
006164  004767  000000G                 CALL    LDCLEN          ;NOW RESET ALL LOGICAL DISK ASSIGNMENTS
006170  000167  000000G                 JMP     RDCMD           ;FINISHED
```

```
                                        ;
                                        ;  SET LD EMPTY
                                        ;  (Dismount all logical disks)
                                        ;
006174                                  SLDEMP:
                                        ;
                                        ;  Begin loop to close log files and deassign any assignments for each
                                        ;  logical disk
                                        ;
006174  005002                                  CLR     R2              ;Init index to 1st LD
006176  004767  000102                  1$:     CALL    LOGDEA          ;Close any log files and make deassignments
006202  062702  000002                          ADD     #2,R2           ;Get index for next LD
006206  020227  000016                          CMP     R2,#14.         ;Have we done all?
006212  101771                                  BLOS    1$              ;Loop if not
                                        ;
                                        ;  Dismount the all LD's
                                        ;
006214  012700  000110'                         MOV     #DMTALD,R0      ;Emt arg. block to dismount all LD's
006220  104375                                  EMT     375             ;Dismount all LD's
006222  103003                                  BCC     9$              ;Branch if all LD's dismounted
006224                                          .PRINT  #EDMALD         ;"Unable to dismount all logical disks."
                                        ;
                                        ;  Finished
                                        ;
006232  000167  0000000                 9$:     JMP     RDCMD
```

```
                                ;-------------------------------------------------------------------------------
                                ;  Subroutine to dismount a logical disk.
                                ;
                                ;  Inputs:
                                ;   R2 = LD unit index
                                ;
006236  010246                  LDDMT:  MOV     R2,-(SP)
006240  004767  000040                  CALL    LOGDEA          ;Close any log files and make deassignments
                                ;
                                ;  Tell system to stop doing directory caching for device
                                ;
006244  112767  000001  000000G         MOVB    #1,SERFLG       ;Do .SERR to avoid abort for illegal device
006252  012700  000000G                 MOV     #DMTARG,R0      ;Point to EMT arg block
006256  104375                          EMT     375             ;Tell system to stop doing caching
006260  105067  000000G                 CLRB    SERFLG          ;Do .HERR
                                ;
                                ;  Clean out LD info tables
                                ;
006264  005062  000000G                 CLR     LDPDEV(R2)      ;No physical device assignment
006270  072227  000002                  ASH     #2,R2           ;Get index into LDNAME table
006274  005062  000000G                 CLR     LDNAME(R2)      ;No file name
                                ;
                                ;  Finished
                                ;
006300  012602                  9$:     MOV     (SP)+,R2
006302  000207                          RETURN
```

```
                                ;-----------------------------------------------------------------------
                                ;
                                ;  Subroutine to check for open log files and deassign any assignments
                                ;
                                ;  Inputs:
                                ;   R2 = LD unit index
                                ;
006304  010246                  LOGDEA: MOV     R2,-(SP)
006306  010546                          MOV     R5,-(SP)
                                ;
                                ;  See if this LD is in use
                                ;
006310  010200                          MOV     R2,R0           ;Get LD unit index
006312  072027  000002                  ASH     #2,R0           ;Cvt to index into name table
006316  005760  000000G                 TST     LDNAME(R0)      ;Is this LD in use?
006322  001413                          BEQ     9$              ;Br if not
                                ;
                                ;  Build name of LD
                                ;
006324  010205                          MOV     R2,R5           ;Get LD unit index
006326  006205                          ASR     R5              ;Convert to unit number
006330  066705  000000G                 ADD     R50LDO,R5       ;Add "LDO" to form unit name
006334  010567  000000G                 MOV     R5,MNTDEV       ;Save name of device being dismounted
                                ;
                                ;  Close the log file if it is on the device being dismounted
                                ;
006340  004767  000000G                 CALL    LOGCHK          ;Close log file if it is on this LD
                                ;
                                ;  Do DEASSIGN of anything assigned to this LD
                                ;
006344  010500                          MOV     R5,R0           ;Get LD name
006346  004767  000000G                 CALL    DEADEV          ;Do deassign
                                ;
                                ;  Finished
                                ;
006352  012605                  9$:     MOV     (SP)+,R5
006354  012602                          MOV     (SP)+,R2
006356  000207                          RETURN
```

```
                                        .SBTTL  .       SL
                                ;-----------------------------------------------------------------------
                                ;  SET SL
                                ;
                                ;  Inputs:
                                ;   R3 = Pointer to parameters following "SET SL".
                                ;
006360                          SETSL:
                                ;
                                ;  Process each of the command parameters
                                ;
006360  012704  001314'                 MOV     #SLEHD,R4       ;Point to list of qualifiers
006364  004767  0000000                 CALL    SCNOPS          ;Process the qualifiers
                                ;
                                ;  Finished
                                ;
006370  000167  0000000                 JMP     RDCMD


                                ;-----------------------------------------------------------------------
                                ;  SET SL ON
                                ;
006374  105767  0000000         SLOON:  TSTB    VSLEDT          ;Is SL available?
006400  001007                          BNE     1$              ;Br if yes
006402                                  FWARN   #EM$NSL         ;SL not genned into system
006416  000425                          BR      9$
006420  032761  000000C 0000000 1$:     BIT     #<VT100!VT200!VT52>,LTRMTP(R1) ;VT100, VT200 or VT52
006426  001007                          BNE     2$              ;Br if yes
006430                                  FWARN   #EM$SLT         ;Invalid terminal type
006444  000412                          BR      9$
006446  052761  0000000 0000000 2$:     BIS     #$SLON,LSW7(R1) ;Enable SL for this line
006454  032761  0000000 0000000         BIT     #$SLKED,LSW7(R1);Is Ked mode wanted?
006462  001403                          BEQ     9$              ;Br if not
006464                                  .PRINT  #SLKDON         ;Enable terminal alternate keypad mode
006472  000207                  9$:     RETURN
                                ;
                                ;-----------------------------------------------------------------------
                                ;  SET SL OFF
                                ;
006474                          SLOOFF: .PRINT  #SLKDOF         ;Return numeric keypad to normal mode
006502  042761  0000000 0000000         BIC     #$SLON,LSW7(R1) ;Disable SL for this line
006510  000207                          RETURN
                                ;
                                ;-----------------------------------------------------------------------
                                ;  SET SL TTY
                                ;
006512  052761  0000000 0000000 SLOTTY: BIS     #$SLTTY,LSW7(R1);Enable SL for .TTYIN
006520  000207                          RETURN
                                ;
                                ;-----------------------------------------------------------------------
                                ;  SET SL NOTTY
                                ;
006522  042761  0000000 0000000 SLONTT: BIC     #$SLTTY,LSW7(R1);Disable SL for .TTYIN
006530  000207                          RETURN
                                ;
                                ;-----------------------------------------------------------------------
                                ;  SET SL WIDTH=n
```

```
                                        ;
006532                                  SLOWID: FWARN   #EM$SLW         ;Print warning message
006546  012705  000000G                        MOV     #SLMXLN,R5      ;Get SL width
006552  004767  000000G                        CALL    PRTDEC          ;Print the value
006556                                          .PRINT  #CRLF           ;Terminate the line
006564  000207                                  RETURN
                                        ;
                                        ;-------------------------------------------------------------------
                                        ;  SET SL LET
                                        ;
006566  052761  000000G 000000G         SLOLET: BIS     #$SLLET,LSW7(R1);Enable LET feature
006574  000207                                  RETURN
                                        ;
                                        ;-------------------------------------------------------------------
                                        ;  SET SL NOLET
                                        ;
006576  042761  000000G 000000G         SLONLT: BIC     #$SLLET,LSW7(R1);Disable LET feature
006604  000207                                  RETURN
                                        ;
                                        ;-------------------------------------------------------------------
                                        ;  SET SL KED
                                        ;
006606  052761  000000G 000000G         SLOKED: BIS     #$SLKED,LSW7(R1);Remember SL is in KED mode
006614                                          .PRINT  #SLKDON         ;Enable terminal alternate keypad mode
006622  000207                                  RETURN
                                        ;
                                        ;-------------------------------------------------------------------
                                        ;  SET SL RT11
                                        ;
006624  042761  000000G 000000G         SLORT:  BIC     #$SLKED,LSW7(R1);Remember SL is not in KED mode
006632                                          .PRINT  #SLKDOF         ;Return numeric keypad to normal mode
006640  000207                                  RETURN
                                        ;
                                        ;-------------------------------------------------------------------
                                        ;  Unimplemented SL option
                                        ;
006642                                  SLOUNI: FWARN   #EM$UIO         ;Unimplemented option
006656  000207                                  RETURN
                                        ;
                                        ;-------------------------------------------------------------------
                                        ;  Ignored option
                                        ;
006660  000207                          SLONOP: RETURN
                                        ;
                                        ;-------------------------------------------------------------------
                                        ;  SET RECALL NORMAL/REVERSE
                                        ;
006662  012704  001444'                 SETREC: MOV     #RCLHD,R4       ;Point to option table
006666  004767  000000G                         CALL    SEARCH          ;Accrue and check options
006672  103006                                  BCC     1$              ;Br if valid option
006674                                          FWARN   #INVOPT         ;Invalid option
006710  111467  000000G                 1$:     MOVB    @R4,RCLREV      ;Set into context cell
006714  000167  000000G                         JMP     RDCMD
                                        
000001                                          .END
Errors detected:  0
```

```
.        SL

*** Assembler statistics


Work  file  reads:  0
Work  file writes:  0
Size of work file:  12080 Words  ( 48 Pages)
Size of core pool:  18176 Words  ( 71 Pages)
Operating  system:  RT-11

Elapsed time:  00:01:56.07
,LP:TSKST1=DK:TSKST1/C/N:SYM
```

Cross reference table (CREF V05.05)


```
$1STLG    1-73
$8BIT     1-111
$AUTO     1-87
$CARUP    1-85
$CCLRN    1-86
$CFABT    1-106
$CFALL    1-112
$CFCCL    1-112
$CFDCC    1-112
$CFOPN    1-118
$CFSOT    1-110
$CHACT    1-61
$CLTST    1-96        5-45        5-46
$CTRLC    1-104
$CTRLD    1-154      10-38       10-40
$CTRLO    1-61
$CTRLS    1-91
$DBKMN    1-84        5-38        5-39
$DEAD     1-158
$DEBUG    1-155
$DEFER    1-124
$DETCH    1-89
$DIBOL    1-73        5-167       5-168
$DILUP    1-108
$DISCN    1-90
$DOOFF    1-114
$DUPRN    1-109
$ECHO     1-111
$EMTTR    1-95        5-174       5-175       10-48
$FORM     1-110
$FORMO    1-112
$HARD     1-158
$HITTY    1-72
$INCOR    1-128
$INDAB    1-159       5-62        5-63
$INDDF    1-157       5-32        5-33        14-86       14-96
$INDRN    1-157
$INIT     1-158
$INKMN    1-104
$KED      1-128      14-43       14-48       14-56
$KINIT    1-68
$LC       1-111
$LOFCF    1-203
$MLOCK    1-77
$NOIN     1-72       17-6
$NOINT    1-204
$NOWTT    1-72
$PAGE     1-111
$PHONE    1-158
$PRGLK    1-87
$PWKEY    1-32       20-103      20-109
$QTSET    1-133
$QUIET    1-125
$RNIOP    1-205
$SCOPE    1-111
$SGALL    1-124       5-206
```

Cross reference table (CREF V05.05)

```
$SGHIO    1-123     5-188     5-189
$SGIIO    1-123     5-190     5-191
$SGQO     1-105     5-192     5-193
$SGQ1     1-123     5-194     5-195
$SGQ1A    1-123     5-196     5-197
$SGQ1B    1-123     5-198     5-199
$SGQ1C    1-123     5-200     5-201
$SGQ2     1-123     5-202     5-203
$SGQ3     1-105     5-204     5-205
$SLKED    1-82     34-32     34-80     34-87
$SLLET    1-202    34-68     34-74
$SLON     1-202    34-31     34-41
$SLTTY    1-202    34-47     34-53
$SNWTT    1-155
$SPLJB    1-91
$STSNG    1-61
$SUCF     1-113    17-7      26-21     27-22
$SYSPS    1-160
$TAB      1-110
$TAPE     1-153
$TECO     1-133    14-43     14-49     14-55
$TTGAG    1-118
$UCLCF    1-119     5-53     14-61     14-68     14-74     14-79
$UCLCL    1-121     5-55     14-62     14-68     14-73     14-79
$UCLCM    1-121     5-54     14-62     14-67     14-74     14-79
$UCLRN    1-120
$UKMON    1-85      5-34     28-45
$VNOTT    1-48
$VTESC    1-81
$WILD     1-134     5-25      5-26
...V1      8-57     9-28      9-30     10-16     10-18     11-28     11-28     11-28     11-30     11-30     11-35     11-41
          11-41     11-44     12-74     12-75     12-75     12-95     12-96     12-96     12-97     12-97     13-5      13-5
          13-6      13-6      13-38     14-87     14-90     14-97     14-100    18-60     18-66     18-72     18-78     21-11
          21-14     21-21     22-11     22-14     23-9      25-28     25-42     25-42     25-42     28-42     28-42     28-42
          28-44     28-44     29-6      29-6      29-6      29-11     29-11     29-30     29-30     29-34     29-35     29-35
          29-50     29-50     29-50     29-52     29-52     31-21     34-34     34-40     34-62     34-81     34-88
...V2      8-57      8-57#    11-28     11-28     11-28#    11-28#    11-30     11-30     11-30#    11-30#    11-35     11-35#
          11-41     11-41     11-41#    11-41#    11-44     11-44#    12-74     12-74#    12-75     12-75     12-75#    12-75#
          12-95     12-95#    12-96     12-96     12-96#    12-96#    12-97     12-97     12-97#    12-97#    13-5      13-5
          13-5#     13-5#     13-6      13-6#     13-38     13-38#    14-90     14-90#    14-90#    14-100    14-100#   14-100#
          23-9      23-9#     23-9#     25-42     25-42     25-42#    25-42#    28-42     28-42     28-42#    28-42#    28-44
          28-44#    29-6      29-6      29-6#     29-6#     29-11     29-11     29-11#    29-11#    29-30     29-30     29-30#
          29-30#    29-34     29-34#    29-35     29-35     29-35#    29-35#    29-50     29-50     29-50#    29-50#    29-52
          29-52     29-52#    29-52#
...V3     14-90     14-90     14-90#    14-100    14-100    14-100#   23-9      23-9      23-9#
ABRTAD    1-48
ABRTCD    1-48
ABRTCF    1-165
ACRDEC    1-173     9-16      9-48      10-8      12-58     14-33     19-8      19-33     21-5      22-5
ACRFIL    1-172    25-16     26-35     28-21
ACRFN     1-165
ACROCT    1-175    12-60
ACRSPD    1-76
ACRSTR    1-37     15-22
ACRTXT    1-39     16-13     27-30
AD$$SZ    1-64
```

Cross reference table (CREF V05.05)


```
AD$DVU    1-64
AD$JOB    1-64
ADM3A     1-148
ADM3FL    1-149
ADM3NO    1-151
AF$DBG    1-204
AF$HIE    1-204
AF$IOP    1-205
AF$MEM    1-47
AF$NOI    1-204
AF$NOW    1-47
AF$PLK    1-204
AF$SCA    1-47
ALCDEV    1-28
ALCEND    1-63
ALCHD1    1-60
ALCHD2    1-60
ALCTBL    1-63
ALDBLK    1-188
ALDEX     1-187     1-188
ALFN      1-199
AMBOPT    1-173     1-186     8-56
AR$$SZ    1-202
AR$CNT    1-201
AR$CON    1-201
AR$CPH    1-201
AR$CPL    1-201
AR$DMY    1-202
AR$PRG    1-201
AR$PRJ    1-201
AR$UNM    1-201
ARNRPB    1-202
ASDEX     1-170
ASKLNM    1-169
ASNHD1    1-182
ASNHD2    1-182
ASNOVF    1-170     1-195
ASNSRC    1-137     7-45
AT$DEV    1-56      7-47
AUTHFN    1-192
BADCMD    1-169
BADPMT    1-176     15-38
BADPRI    1-176     21-8      22-8
BADSAV    1-169
BASMAP    1-142
BDFNAM    1-172     25-80     26-57     27-50     28-59
BDLGOP    1-177     25-44
BDLIN     1-190
BDSO.     8-44      8-54#     26-13
BELL      2-7#
BLANK     2-6#
BLKO      1-168     11-30     12-23     12-39     13-5      13-28*    13-29*    13-30     15-25     16-23     27-37     29-11
          29-12     29-25     29-26
BLKWDS    2-11#
BOTDEV    1-103     23-18*
BOTUNI    1-103
```

Cross reference table (CREF V05.05)

```
C.CSW     1-115
C.DEVQ    1-115
C.SBLK    1-115
C.USED    1-41      24-34*
CALUCL    1-181
CASCBR    1-97
CASCUP    1-97
CASTBR    1-97
CASTBW    1-97
CASTRO    1-98
CASTWO    1-98
CCLHD     4-10       5-44#
CCLNAM    1-196     29-50
CCLSAV    1-94      29-52
CD$$SZ    1-116
CD$$UB    1-116
CD$BAS    1-116
CD$DVU    1-116
CD$JOB    1-116
CD$NAM    1-116
CD$TOP    1-133
CDBUF     1-38
CDGET     1-38
CF$IND    1-159     14-89      14-99
CF$QUT    1-159
CFABLV    1-160     23-9
CFBLK     1-125
CFBUF     1-94
CFCHAN    1-124
CFEND     1-94
CFHOLD    1-132
CFIND     1-114
CFLFL4    1-119
CFNEST    1-117
CFPNT     1-125
CFSEND    1-118
CFSP      1-118
CFSPND    1-131
CFSTK     1-68
CFSTS     1-159     14-87      14-90      14-97      14-100
CHAIN     1-104
CHKALC    1-111      7-75
CHKCLU    1-36       7-65
CHKDEV    1-196     25-49
CHKDLM    1-186
CHKEQ     1-45      20-20      20-32
CHKMNT    1-171
CHKMTX    1-171
CHKTTD    1-194     25-37
CHNSIZ    1-41      24-32
CINDAT    1-141
CINFLG    1-48
CKACOJ    1-45
CKCLUS    1-37
CKPRIV    1-173     14-6       14-14      23-16      23-24
CKSYPV    1-45       8-14       9-12       9-44      10-4       16-9       29-5       29-49
```

Cross reference table (CREF V05.05)

```
CKTERM   1-174
CL$COL   1-66
CL$EPN   1-41
CL$EPS   1-41
CL$LEN   1-101
CL$LIN   1-101
CL$LIX   1-62
CL$OPT   1-100
CL$RQH   1-63
CL$SKP   1-101
CL$WID   1-101
CL$WQH   1-63
CL$XLN   1-37
CLDEVX   1-139
CLEOFS   1-41
CLFREE   1-57
CLLINE   1-57
CLOTIR   1-65
CLRBIT   5-26    5-34    5-39    5-46    5-63    5-168   5-175   5-189   5-191   5-193   5-195   5-197
         5-199   5-201   5-203   5-205   5-206   14-15#
CLRPRV   1-43
CLSFEP   1-42
CLSFSP   1-98
CLTOTL   1-98    7-63
CLUNIT   1-57
CMDBUF   1-164   1-181
CMDCCL   1-196
CMDDSN   1-27
CMDFRM   1-27
CMDHD    1-26
CMDMEM   1-28
CMDOFF   1-26
CMDSET   1-28    7-5#
CMDSHO   1-28
CMDUSE   1-28
CMDWHO   1-28
CO$8BT   1-115
CO$BNI   1-100
CO$BNO   1-100
CO$CR    1-99
CO$CTL   1-99
CO$DEF   1-66
CO$DTR   1-98
CO$FF    1-99
CO$FFO   1-99
CO$LC    1-99
CO$LFI   1-100
CO$LFO   1-100
CO$TAB   1-99
COAD     1-187   1-188
COAF     1-200
COAL     1-187   1-188
COLOO    1-181
COMMA    2-10#
CONFG2   1-62
CONFIG   1-113
```

Cross reference table (CREF V05.05)

```
CORUSR    1-61      14-34      20-102     20-108     22-20
CPUAH     1-181
CPUAL     1-181
CR        2-5#
CRLF      1-176      9-30      10-18      21-14      21-21      22-14      34-62
CS$RON    1-80
CSHALC    1-103     10-9
CSHDEV    1-108
CSHDVN    1-108
CSHHD     1-163
CSHMSG    1-203
CSHSIZ    1-143
CSIMS1    1-175     13-39      30-17
CSIMS2    1-173      7-44      11-36
CSIMS4    1-195
CTDHD     4-12       5-236#
CTRLTT    1-102
CURMTX    1-186
CURPRM    1-130
CVDVNM    1-183
CVTTAB    1-164      7-5
CVTUC     1-163
CW$SOH    1-113
CW$PRO    1-62
CXTBAS    1-107
CXTPAG    1-79      29-23
CXTRMN    1-41      24-26
CXTWDS    1-107
DATTIM    1-27
DCCRD     1-137
DCCWR     1-137
DCHNEW    6-33#     10-26
DCTRD     1-137
DCTWR     1-137
DEADEV    1-171     33-32
DELSPC    1-185
DETARG    1-200
DETHD     1-200
DETTXT    1-179
DEVHD1    1-182
DEVIDL    1-187      1-187      1-188
DEVUNT    1-174     11-20*     12-85
DFJMEM    1-68
DIABFL    1-125
DIABLO    1-147
DIABNO    1-126
DIVIDE    1-180
DIVSOR    1-193
DJABMS    1-191
DKASHD    1-59
DKSAV     1-164
DLCEMT    1-28
DLMSG     1-191
DLTXT     1-178
DMTALD    6-60#     31-18
DMTALL    1-191
```

Cross reference table (CREF V05.05)

```
DMTARG    1-170     32-13
DMTSUB    1-196
DOASGN    1-85
DORUN     1-27
DOSTOP    1-192
DS$DIR    1-204     25-26      25-29
DSBLOK   25-26*     25-28      25-29      25-81#
DVFLAG    1-32      13-22
DVSHH1    1-59
DVSHH2    1-59
DVSHH3    1-59
DVSTAT    1-76
DX$NST    1-32      13-22
DZTXT     1-198
EDIT      1-73
EDITHD    4-13       5-4#
EDMALD    1-42      31-21
EDTFIL    1-184
EM$ACL    1-54
EM$CAP    1-44
EM$CIP    1-53      22-18
EM$CLB    1-54
EM$CLN    1-53
EM$CLX    1-36
EM$CNO    1-44
EM$CPO    1-44
EM$CSE    1-70
EM$HNI    1-68      13-16
EM$ICL    1-52
EM$IDR    1-42
EM$ILN    1-53       1-54
EM$IST    1-42
EM$IUN    1-53
EM$NAD    1-46      10-37      10-47
EM$NPD    1-53      10-34
EM$NPR    1-45
EM$NSF    1-53      26-25      27-26
EM$NSL    1-54      34-25
EM$NUK    1-86      28-51
EM$OPR    1-36      11-45
EM$PTA    1-36      20-41
EM$PTU    1-36      20-43
EM$SLT    1-54      18-46      34-29
EM$SLW    1-54      34-59
EM$SPL    1-39      16-19
EM$STL    1-42      27-33
EM$TSL    1-54
EM$UIO    1-55      34-94
EM$WC0    1-40      18-96
EM$WC1    1-40      18-97
EM$WC2    1-40      18-98
EM$WC3    1-40      18-99
EM$WCM    1-40      18-11
EMTHD     4-14       5-173#
ERRHD     4-16       5-13#
ERRLOC    1-67      18-86
```

Cross reference table (CREF V05.05)

```
ERRSEV    1-134    23-5*
ESC       1-82
FC$CDX    1-163
FC$LNK    1-163
FD$NAM    1-163
FF        2-9#
FILNAM    1-165     1-166    25-21    25-28    25-31    25-36    25-42    25-42    25-48    26-40    26-45    28-23
FKILL     1-165     7-44     8-56     8-58    10-34    10-37    10-47    11-36    11-45    13-33    13-39    14-39
         14-92    16-19    18-11    18-46    18-92    22-18    25-44    25-80    26-25    26-57    27-26    27-33
         27-50    28-51    28-55    28-59    29-43    29-54    30-17
FPRINT    1-165    15-38    20-41    20-43    21-8     21-18    22-8
FSTDL     1-89
FSTIOL    1-62      1-79
GAGMSG    1-190
GENTOP    1-103
GETKCH    1-77     12-8     12-11
GRT1      1-144
GTRD50    1-170     7-40     8-28    12-19    20-21
H.CSR     1-76
H.VEC     1-76
HANBSY    1-175    13-33
HANCHN    1-64      8-57     8-57    11-28    11-28    11-30    11-30    11-35    11-35    11-41    11-41    11-44
         11-44    12-97    12-97    13-5     13-5     13-6     13-6     13-38    13-38
HANENT    1-75
HANPAR    1-76
HANSAV    6-4#     11-41    12-75    12-97
HANSIZ    1-75
HAZEL     1-77
HAZLFL    1-77
HAZLNO    1-77
HIMAP     1-142
HIPRI     1-176    21-18
HNBUF     1-174    11-27*    11-28     13-11
HUPARG    1-195    13-26
II$$SZ    1-46
II$FLG    1-46
II$NAM    1-46
II$NPV    1-48
II$PRV    1-48
IIBUF     1-46
ILLCMD    1-171
ILSW2     1-83
IN$ACT    1-157
IN$CMD    1-157
IN$CNT    1-157
INDACT    1-167
INDERR    1-106
INDHD     4-20      5-61#
INDSAV    1-157    14-84
INDSTA    1-106
INFOMT    1-171
INGADR    1-46
INGEMT    1-46
INSTBL    1-46
INSTBN    1-47
INVDAT    1-200
```

Cross reference table (CREF V05.05)

```
INVDEV    1-199
INVEC     1-158
INVLDM    1-195
INVLDN    1-172
INVLDO    8-57#     12-26
INVOPT    1-165      1-171      1-186      8-58      34-108
INVSOP    8-55       8-58#
INVTIM    1-191
IOABFL    1-61       5-181      5-182
IOABHD    4-22       5-180#
ITRMTP    1-159
JCXPGS    1-142
JCXSMS    1-198
JPWDEV    1-39      20-25*
JPWFLG    1-39      20-19*     20-37*     20-52*     20-57*     20-62*     20-67*     20-72*     20-77*     20-82*     20-87*     20-92*
         20-97*
JPWTYP    1-39      20-36*
JSTKND    1-102
JSWLOC    1-67
K52       1-73
KBMSG     1-180
KBTX      1-184
KCSIBF    1-169
KCSIMS    1-170
KDOCIN    1-26
KED       1-73
KILEMT    1-192
KL3CLR    1-87
KL4CLR    1-127
KMNBAS    1-154     29-16
KMNCHN    1-94      12-96      29-30      29-35
KMNHI     1-78      29-17*
KMNNAM    1-196     29-6
KMNPGS    1-79      29-24*
KMNSTK    1-79      29-25*
KMNSTR    1-79      29-26*
KMNTOP    1-79      29-15*
KMONHD    4-23       5-31#
KMPRMT    1-138     15-10
L         4-9        4-9#       4-10       4-10#      4-11       4-11#      4-12       4-12#      4-13       4-13#      4-14       4-14#
          4-15       4-15#      4-16       4-16#      4-17       4-17#      4-18       4-18#      4-19       4-19#      4-20       4-20#
          4-21       4-21#      4-22       4-22#      4-23       4-23#      4-24       4-24#      4-25       4-25#      4-26       4-26#
          4-27       4-27#      4-28       4-28#      4-29       4-29#      4-30       4-30#      4-31       4-31#      4-32       4-32#
          4-33       4-33#      4-34       4-34#      4-35       4-35#      4-36       4-36#      4-37       4-37#      4-38       4-38#
          4-39       4-39#      4-40       4-40#      4-41       4-41#      4-42       4-42#      4-43       4-43#      4-44       4-44#
          4-45       4-45#      4-46       4-46#      4-47       4-47#      4-48       4-48#      4-49       4-49#      4-50       4-50#
          4-51       4-51#      4-52       4-52#      4-53       4-53#      4-54       4-54#      4-55       4-55#      4-56       4-56#
          4-57       4-57#      4-58       4-58#      4-59       4-59#      4-60       4-60#      4-61       4-61#      4-62       4-62#
          4-63       4-63#      5-5        5-5#       5-6        5-6#       5-7        5-7#       5-8        5-8#       5-14       5-14#
          5-15       5-15#      5-16       5-16#      5-17       5-17#      5-18       5-18#      5-19       5-19#      5-25       5-25#
          5-26       5-26#      5-32       5-32#      5-33       5-33#      5-34       5-34#      5-35       5-35#      5-36       5-36#
          5-37       5-37#      5-38       5-38#      5-39       5-39#      5-45       5-45#      5-46       5-46#      5-47       5-47#
          5-53       5-53#      5-54       5-54#      5-55       5-55#      5-56       5-56#      5-62       5-62#      5-63       5-63#
          5-69       5-69#      5-70       5-70#      5-71       5-71#      5-72       5-72#      5-73       5-73#      5-79       5-79#
          5-80       5-80#      5-81       5-81#      5-82       5-82#      5-83       5-83#      5-84       5-84#      5-85       5-85#
          5-86       5-86#      5-87       5-87#      5-88       5-88#      5-89       5-89#      5-90       5-90#      5-91       5-91#
```

Cross reference table (CREF V05.05)

	5-92	5-92#	5-93	5-93#	5-94	5-94#	5-95	5-95#	5-96	5-96#	5-97	5-97#
	5-98	5-98#	5-99	5-99#	5-105	5-105#	5-106	5-106#	5-112	5-112#	5-113	5-113#
	5-114	5-114#	5-115	5-115#	5-116	5-116#	5-117	5-117#	5-118	5-118#	5-119	5-119#
	5-120	5-120#	5-121	5-121#	5-122	5-122#	5-123	5-123#	5-129	5-129#	5-130	5-130#
	5-131	5-131#	5-132	5-132#	5-133	5-133#	5-134	5-134#	5-135	5-135#	5-136	5-136#
	5-137	5-137#	5-138	5-138#	5-139	5-139#	5-140	5-140#	5-141	5-141#	5-142	5-142#
	5-148	5-148#	5-149	5-149#	5-150	5-150#	5-151	5-151#	5-152	5-152#	5-153	5-153#
	5-154	5-154#	5-155	5-155#	5-156	5-156#	5-157	5-157#	5-158	5-158#	5-159	5-159#
	5-160	5-160#	5-161	5-161#	5-167	5-167#	5-168	5-168#	5-174	5-174#	5-175	5-175#
	5-181	5-181#	5-182	5-182#	5-188	5-188#	5-189	5-189#	5-190	5-190#	5-191	5-191#
	5-192	5-192#	5-193	5-193#	5-194	5-194#	5-195	5-195#	5-196	5-196#	5-197	5-197#
	5-198	5-198#	5-199	5-199#	5-200	5-200#	5-201	5-201#	5-202	5-202#	5-203	5-203#
	5-204	5-204#	5-205	5-205#	5-206	5-206#	5-212	5-212#	5-213	5-213#	5-214	5-214#
	5-215	5-215#	5-216	5-216#	5-217	5-217#	5-218	5-218#	5-219	5-219#	5-225	5-225#
	5-231	5-231#	5-237	5-237#	5-238	5-238#						
LA120	1-147											
LA12FL	1-148											
LA12NO	1-148											
LA36	1-147											
LA36FL	1-126											
LA36NO	1-126											
LACTIV	1-80											
LAFSIZ	1-83											
LANGHD	4-24	5-166#										
LBSPRI	1-160	22-21										
LCBIT	1-147											
LCDTYP	1-66											
LCLTXT	1-57											
LCLUNT	1-62											
LCOL	1-82	1-133										
LCONTM	1-91											
LCPUHI	1-91											
LCPULO	1-91											
LD$RON	1-121	30-30	30-35									
LDBASE	1-122	25-59										
LDCLEN	1-71	30-25	30-41									
LDDEVX	1-124	25-51										
LDDMT	30-40	32-7#										
LDFLAG	1-122	30-30*	30-35*									
LDMNT	1-70											
LDNAM	1-169											
LDNAME	1-122	32-21*	33-15									
LDOPHD	1-177	5-68#	30-14									
LDPDEV	1-122	25-58	32-19*									
LDSIZE	1-122											
LF	2-4#											
LF$IN	1-161	24-51	24-56	24-62	25-68							
LF$OPN	1-162	24-21	25-68									
LF$OUT	1-161	24-50	24-57	24-62	25-68							
LF$WRT	1-162	24-40	24-45	25-68								
LFWLIM	1-83											
LICTXT	1-144											
LINBUF	1-78											
LINCNT	1-80											
LINCUR	1-83											
LINFRE	1-191											

Cross reference table (CREF V05.05)

```
LINIR     1-65
LINNXT    1-78
LINPNT    1-80
LINRTS    1-65
LITIME    1-105
LJSW      1-102
LMXLN     1-158
LMXNUM    1-156
LMXPRM    1-159
LNAME     1-60
LNBLKS    1-107
LNMAP     1-113
LNPRIM    1-113
LNSBLK    1-108
LNSPAC    1-117
LOCKTX    1-180
LOFSPC    1-110     26-46
LOGASN    1-172     25-22      26-41      28-25
LOGBAS    1-132      1-134     25-55*     25-59*
LOGBLK    1-161     24-24*     25-70*
LOGBUF    1-161     24-23      25-69
LOGCHK    1-133     33-27
LOGCHN    1-161     24-25      25-42      25-42
LOGCLS    1-176     24-15      25-6
LOGDEA   31-11      32-8       33-8#
LOGDVU    1-132      1-134     25-53*     25-54*     25-58*
LOGFLG    1-161     24-21      24-40*     24-45*     24-50*     24-51*     24-56*     24-57*     24-62*     25-68*
LOGPTR    1-161     24-23*     25-69*
LOMAP     1-142
LOTBUF    1-81
LOTNXT    1-81
LOTPNT    1-81
LOTSIZ    1-82
LOTSPC    1-82
LOUTIR    1-65
LP$7BT    1-52
LP$ODD    1-52
LP$PAR    1-52
LP$SPD    1-52
LPRG1     1-145
LPRG2     1-145
LPRI      1-160
LPROG     1-90
LPROJ     1-90
LRBFIL    1-114
LRDTIM    1-80
LSCCA     1-110
LSECPT    1-95
LSPEMT    6-27#
LSTACT    1-78
LSTATE    1-140
LSTDL     1-89
LSTHL     1-62
LSTIOL    1-62
LSTMX     1-156
LSTPL     1-135
```

Cross reference table (CREF V05.05)

```
LSTPRM    1-131
LSTSL     1-140
LSTSPL    1-76
LSUCF     1-86
LSW       1-61
LSW11     1-32     20-103*   20-109*
LSW2      1-104
LSW2S     1-109
LSW3      1-109    17-6*
LSW4      1-127
LSW5      1-87      5-25      5-26      5-32      5-33      5-45      5-46     14-43*    14-48*    14-49*    14-55*    14-56*
         14-86*    14-96*
LSW6      1-155     5-167     5-168     5-174     5-175    10-48*
LSW7      1-159     5-34      5-53      5-54      5-55      5-56      5-62      5-63     14-61*    14-62*    14-67*    14-68*
         14-73*    14-74*    14-79*    28-45*    34-31*    34-32     34-41*    34-47*    34-53*    34-68*    34-74*    34-80*
         34-87*
LSW8      1-123     5-188     5-189     5-190     5-191     5-192     5-193     5-194     5-195     5-196     5-197     5-198
          5-199     5-200     5-201     5-202     5-203     5-204     5-205     5-206
LSW9      1-85      5-38      5-39     10-38*    10-40*    17-7*     26-21     27-22
LTRMTP    1-148    18-44     34-27
LTSCMD    1-117
LUNAME    1-90
MAXALC    1-63
MAXASN    1-106
MAXAVL    1-174     9-28      10-16
MAXMEM    1-67
MAXMTX    1-186
MAXPRI    1-67      1-160     21-6      22-6      22-12
MAXSEC    1-95
MAXWEM   18-87     18-100#
MDT       1-77
MHNSIZ    1-96
MHNSMS    1-97
MINTIM    1-95
MISSEQ    1-175    14-39     28-55
MNBASE    1-185
MNBPC     1-184
MNFLGS    1-184
MNTARG    1-195
MNTDEV    1-170    33-23*
MNTFUL    1-172
MNTOP     1-185
MNTTXT    1-197
MONAR1    1-185
MONAR2    1-185
MONHD     1-185
MONVEC    1-160    14-87     14-90     14-97     14-100    23-9
MSGBUF    1-190
MSGEND    1-190
MTOPHD    1-171
MUL32     1-200
MXCSR     1-156
MXDTR     1-156
MXJADR    1-69
MXJMEM    1-68
MXJPRI    1-160    21-12     21-16     21-19     21-22     22-16     22-19*
```

Cross reference table (CREF V05.05)

```
MXPRMT   1-138    15-36
MXVEC    1-158
NAMTOP   1-144
NARGS    4-8#     4-9      4-9      4-10     4-10     4-11     4-11     4-12     4-12     4-13     4-13     4-14
         4-14     4-15     4-15     4-16     4-16     4-17     4-17     4-18     4-18     4-19     4-19     4-20
         4-20     4-21     4-21     4-22     4-22     4-23     4-23     4-24     4-24     4-25     4-25     4-26
         4-26     4-27     4-27     4-28     4-28     4-29     4-29     4-30     4-30     4-31     4-31     4-32
         4-32     4-33     4-33     4-34     4-34     4-35     4-35     4-36     4-36     4-37     4-37     4-38
         4-38     4-39     4-39     4-40     4-40     4-41     4-41     4-42     4-42     4-43     4-43     4-44
         4-44     4-45     4-45     4-46     4-46     4-47     4-47     4-48     4-48     4-49     4-49     4-50
         4-50     4-51     4-51     4-52     4-52     4-53     4-53     4-54     4-54     4-55     4-55     4-56
         4-56     4-57     4-57     4-58     4-58     4-59     4-59     4-60     4-60     4-61     4-61     4-62
         4-62     4-63     4-63     5-4#     5-5      5-5      5-6      5-6      5-7      5-7      5-8      5-8
         5-13#    5-14     5-14     5-15     5-15     5-16     5-16     5-17     5-17     5-18     5-18     5-19
         5-19     5-24#    5-25     5-25     5-26     5-26     5-31#    5-32     5-32     5-33     5-33     5-34
         5-34     5-35     5-35     5-36     5-36     5-37     5-37     5-38     5-38     5-39     5-39     5-44#
         5-45     5-45     5-46     5-46     5-47     5-47     5-52#    5-53     5-53     5-54     5-54     5-55
         5-55     5-56     5-56     5-61#    5-62     5-62     5-63     5-63     5-68#    5-69     5-69     5-70
         5-70     5-71     5-71     5-72     5-72     5-73     5-73     5-78#    5-79     5-79     5-80     5-80
         5-81     5-81     5-82     5-82     5-83     5-83     5-84     5-84     5-85     5-85     5-86     5-86
         5-87     5-87     5-88     5-88     5-89     5-89     5-90     5-90     5-91     5-91     5-92     5-92
         5-93     5-93     5-94     5-94     5-95     5-95     5-96     5-96     5-97     5-97     5-98     5-98
         5-99     5-99     5-104#   5-105    5-105    5-106    5-106    5-111#   5-112    5-112    5-113    5-113
         5-114    5-114    5-115    5-115    5-116    5-116    5-117    5-117    5-118    5-118    5-119    5-119
         5-120    5-120    5-121    5-121    5-122    5-122    5-123    5-123    5-128#   5-129    5-129    5-130
         5-130    5-131    5-131    5-132    5-132    5-133    5-133    5-134    5-134    5-135    5-135    5-136
         5-136    5-137    5-137    5-138    5-138    5-139    5-139    5-140    5-140    5-141    5-141    5-142
         5-142    5-147#   5-148    5-148    5-149    5-149    5-150    5-150    5-151    5-151    5-152    5-152
         5-153    5-153    5-154    5-154    5-155    5-155    5-156    5-156    5-157    5-157    5-158    5-158
         5-159    5-159    5-160    5-160    5-161    5-161    5-166#   5-167    5-167    5-168    5-168    5-173#
         5-174    5-174    5-175    5-175    5-180#   5-181    5-181    5-182    5-182    5-187#   5-188    5-188
         5-189    5-189    5-190    5-190    5-191    5-191    5-192    5-192    5-193    5-193    5-194    5-194
         5-195    5-195    5-196    5-196    5-197    5-197    5-198    5-198    5-199    5-199    5-200    5-200
         5-201    5-201    5-202    5-202    5-203    5-203    5-204    5-204    5-205    5-205    5-206    5-206
         5-211#   5-212    5-212    5-213    5-213    5-214    5-214    5-215    5-215    5-216    5-216    5-217
         5-217    5-218    5-218    5-219    5-219    5-224#   5-225    5-225    5-230#   5-231    5-231    5-236#
         5-237    5-237    5-238    5-238
NATXT    1-178
NEDCHR   1-65
NEWJSW   1-102
NFRESB   1-129
NINTXT   1-178
NLINES   1-115
NOCCL    1-177    29-54
NOCIN    1-169
NODAT    1-183
NOEQL    14-29    14-39#
NOFIL    1-181     1-189
NOFLAG   1-171     7-8*     12-4*    12-14*   12-41
NOFRDL   1-200
NOIND    1-175    14-92
NOKMON   1-196    29-43
NOLDMT   1-183
NONEMS   1-183
NOOPTT   1-189
NOPMGN   1-185
```

Cross reference table (CREF V05.05)

```
NOPRG     1-165      1-169
NOSTRT    1-168
NOTAVL    1-184
NOTON     1-190
NOTXT     1-178
NOUDC     1-182
NSPLDV    1-121
NSWPMS    1-186
NUCHN     1-114
NUMDCD    1-138      4-28
NUMDEV    1-151     13-10
NUMON     1-83
NXTOPT   11-43      12-4#      12-109
OCTFIX    1-178
OCTPRT    1-198
ODTBAS    1-154
OF$$SZ    1-153
OF$DEV    1-152
OF$FIL    1-152
OF$FLG    1-152
OF$UNT    1-152
OFFEMT    1-192
OKFEND    1-96
OKFILE    1-96
OPRTXT    1-57
OPTLST    1-43
OT$RON    1-153
OTHRON    1-194
OTRMNT    1-196
OVRCOR    1-168
PO$DBG    1-47       10-35      10-45
PO$NAM    1-45
PO$OPR    1-45       11-42
PO$SPV    1-43
PO$SYS    1-34       26-23      27-24
P2$CGR    1-40       18-9
P2$TRM    1-45
PA$BEL    1-36       20-62      20-67
PA$BLD    1-35        5-156      5-157      5-158      5-159      5-160      5-161
PA$DSC    1-35        5-157      5-158
PA$DTS    1-34       20-82      20-87
PA$DWD    1-35        5-155      5-156      5-157      5-158      5-161
PA$FLG    1-34       20-19      20-72      20-77
PA$GRC    1-35        5-155      5-156      5-157      5-158      5-161
PA$HQL    1-35        5-155      5-156      5-157      5-158
PA$LET    1-35       20-52      20-57
PA$NWD    1-36       20-92      20-97
PA$UKC    1-35        5-155      5-156      5-157      5-158
PA$ULN    1-35        5-155      5-156      5-157      5-158      5-159      5-160      5-161
PASLIN    1-134
PAUMSG    1-164
PBFEND    1-118
PEKADR    1-38
PEKEMT    1-38
PEKSIZ    1-38
PF$IOW    1-155
```

Cross reference table (CREF V05.05)

```
PF$SYS    1-155
PFCO      1-43
PFSO      1-43
PHYMEM    1-101
PMBUSY    1-185
PNAME     1-151      1-174     13-12
POPCF     1-175
PRGALL    1-27
PRGSIZ    1-78
PRGTOP    1-78
PRIEMT    6-38#     21-23     22-23
PRIVAO    1-44
PRIVCO    1-47      10-35     10-45     11-42     26-23     27-24
PRIVC2    1-45      18-9
PRIVSO    1-43
PRMBUF    1-131
PRMEND    1-131
PRMPNT    1-130
PROSLT    1-52
PRSBEL    5-129     20-62#
PRSDEV    5-131     20-18#
PRSDRF    5-132     20-57#
PRSFLG    5-133     20-72#
PRSHD     5-128#    20-9
PRSKEY    5-135     20-102#
PRSLET    5-137     20-52#
PRSNBL    5-130     20-67#
PRSNFL    5-134     20-77#
PRSNKY    5-136     20-108#
PRSNST    5-139     20-87#
PRSNWD    5-142     20-97#
PRSSTM    5-138     20-82#
PRSTYP    5-140     20-31#
PRSWID    5-141     20-92#
PRTBUF    1-183
PRTDAT    1-199
PRTDC2    1-180
PRTDC3    1-180
PRTDEC    1-174      9-29      10-17     21-10     21-13     21-20     22-10     22-13     34-61
PRTFIX    1-177
PRTFNM    1-183
PRTLN     1-179
PRTPCT    1-193
PRTR50    1-199
PRTSPC    1-177
PRTTIM    1-199
PRTTMD    1-182
PRTTMV    1-181
PRTTOD    1-199
PRTTTP    1-178
PRTUNM    1-179
PRTWRN    1-203     13-16     13-24     34-25     34-29     34-59     34-94     34-108
PRVCLR   14-14#
PRVLST    1-47
PRVOPT    1-43
PRVSET   14-6#
```

Cross reference table (CREF V05.05)


```
PUSHCF    1-166
PVNPW     1-43
PVON      1-87
PWTHD     5-147#    20-33
QHDMS1    1-59
QHDMS2    1-59
QUME      1-147
QUMEFL    1-149
QUMENO    1-149
R$CHN     1-41      24-27     24-31
R$XCHN    1-41      24-31
R50BUF    1-170      7-39*     7-41      7-42      7-48*     7-67      8-29      11-6      11-11*    11-27     12-27     12-29
          20-25
R50C10    6-6#
R50CLO    6-5#
R50COM    1-110     26-33
R50DIR    1-166
R50DK     1-199
R50DSK    1-172
R50DUP    1-167
R50IND    1-166
R50K52    1-167
R50KED    1-167
R50KEX    1-167
R50KMN    1-37
R50LD     1-172      7-52     30-6
R50LDO    1-170     30-9      33-22
R50LD7    1-172      7-54
R50LOG    1-176     25-14
R50MON    1-200
R50NO     1-173      8-29
R50PIP    1-167
R50SAV    1-166     28-19
R50SY     1-166
R50TSX    1-167
R50TT     1-195
R50UCL    1-167
R50VIR    1-168
RC$$SZ    1-51
RC$AEP    1-50
RC$AGE    1-50
RC$CNT    1-49
RC$EXC    1-50
RC$FLG    1-50
RC$GBL    1-50
RC$LCG    1-52
RC$LEN    1-51
RC$NAM    1-50
RC$OWN    1-49
RC$PVT    1-51
RC$USE    1-50
RCBBAS    1-51
RCBEND    1-51
RCLHD     5-104#    34-105
RCLREV    1-34      34-109*
RDB       1-139
```

Cross reference table (CREF V05.05)

```
RDBEND    1-139
RDCMD     1-164     4-17      4-58      9-40      10-28     10-39     10-41     10-49     13-34     14-10     14-18     14-24
         14-37     14-44     14-50     14-57     14-63     14-69     14-75     14-80     14-91     14-101    15-50     16-30
         17-8      18-82     20-14     21-26     22-26     23-10     23-19     23-26     24-11     26-53     27-13     28-46
         29-53     30-26     30-31     30-36     30-42     31-25     34-17     34-110
RDERM     1-168
REMTXT    1-57
RESDEV    1-153
RJEMT     1-37
RNMS      1-179
RONTXT    1-184
RSR       1-156
RSTIND    5-33      14-96#
RSTPRV    1-44
RT$$SZ    1-139
RT$NAM    1-139
RUNDEV    1-107
RUNEMT    1-168
RUNHD     1-164
RUNMS     1-200
S$HICP    1-92
S$INWT    1-93
S$IOFN    1-88
S$IOWT    1-145
S$LOW     1-90
S$MSWT    1-94
S$OTFN    1-88
S$OTLO    1-88
S$OTWT    1-93
S$QMIO    1-86
S$QUSR    1-145
S$RT      1-90
S$SFWT    1-93      1-145
S$SPCB    1-146
S$SPDB    1-146
S$SPND    1-87
S$TMWT    1-93
S$TTFN    1-88
S$TTSC    1-89
S$TWFN    1-88
S9600     1-93
SBPFIL    5-231     27-17#
SBPHD     5-230#    27-11
SBPSUF    1-36      27-38
SC$ERR    1-33       5-15
SC$FTL    1-33       5-17
SC$NON    1-33       5-19
SC$SEV    1-33       5-16
SC$SUC    1-33
SC$UNC    1-33       5-18
SC$WRN    1-33       5-14
SCHAIN    1-124
SCNOPS    1-37      18-22     20-10     24-10     27-12     34-13
SD$BAK    1-127
SD$DEL    1-119
SD$FLK    1-120
```

Cross reference table (CREF V05.05)

```
SD$HLD    1-130
SD$SNG    1-129
SD$WFM    1-120
SDBLK     1-121
SDBU      1-127
SDBUF1    1-121
SDCB      1-135
SDCBND    1-135
SDCBSZ    1-140
SDDVU     1-139
SDFHD     1-132
SDFLAG    1-120
SDFORM    1-120
SDNAME    1-140
SDSFCB    1-119
SDSKIP    1-127
SEARCH    1-164     7-27      8-43      20-34     26-11     30-15     34-106
SEEDIT    5-5      14-43#
SEK52     5-7      14-54#
SEKED     5-6      14-55#
SERFLG    1-61     32-12*    32-15*
SETBIT    5-25      5-38      5-45      5-62      5-167     5-188     5-190     5-192     5-194     5-196     5-198     5-200
          5-202     5-204    14-7#
SETBYT   14-28#
SETCKP   11-31     11-41#
SETCL     1-32      7-70
SETCNW    5-47     29-49#
SETDEV    7-76     11-6#
SETECO    5-8      14-48#
SETEMT    5-174    10-45#
SETESU    4-15     17-6#
SETHD     1-173     4-8#      7-26
SETHST    1-32      4-19
SETIND    5-32     14-84#
SETIVD   11-15     11-17     11-29     11-35#
SETIVS   12-45     12-64     12-99     13-38#
SETKNW    5-37     29-5#
SETLD     7-58     30-5#
SETLOF    4-26     26-6#
SETLOG    4-25     24-5#
SETMPR    4-27     22-5#
SETNDC    4-28      9-44#
SETNSH    4-46     23-24#
SETOPT   12-23#
SETPRC    1-34      4-32
SETPRI    4-34     21-5#
SETPRS    4-35      4-36      4-63      20-5#
SETPRT    4-33     15-5#
SETPRV    4-22      8-14#
SETREC    4-44     34-105#
SETSBP    4-49     27-7#
SETSEV    5-14      5-15      5-16      5-17      5-18      5-19      23-5#
SETSHT    4-45     23-16#
SETSL     4-48     34-8#
SETSUB    4-10      4-12      4-13      4-14      4-16      4-20      4-23      4-24      4-47      4-57      4-61      8-15#
SETSVB    4-31      9-9#
```

Cross reference table (CREF V05.05)

```
SETSVC    9-10      9-12#
SETSVL    4-11      4-18      4-21      4-29      4-30      4-37      4-38      4-39      4-40      4-41      4-42      4-43
          4-54      4-55      4-56      9-11#
SETSVV    9-22#     9-50
SETSYP    4-50      4-51      16-5#
SETTTY    1-34      4-52      4-53
SETUKM    5-35      5-36      28-7#
SETUP    12-105    13-5#
SETWIN    4-62      18-5#
SETWRD    1-42
SF$1ST    1-130
SF$BSY    1-129
SF$HLD    1-130
SFFILE    1-146
SFFLAG    1-132
SFFORM    1-129
SFID      1-76
SFNMBL    1-129
SFQLNK    1-132
SFUSER    1-146
SH$$SZ    1-74      12-31
SH$FLG    1-74      12-43     12-48     12-56
SH$NAM    1-74      12-27     12-29
SH$RTN    1-74      12-36
SH$VAL    1-74      12-24     12-84
SHMTH1    1-182
SHMTH2    1-182
SHOHD     1-197
SHRRCB    1-51
SHRRCN    1-51
SHTMSG    1-192
SIGHD     4-47      5-187#
SIZEMT    1-195
SIZVAL    1-169
SJEMT     1-37
SKPSPC    1-26      7-6       8-20      8-35      12-5      12-15     12-50     12-54     12-103    15-9      19-27     25-7
         26-29     28-11
SLDCLN    5-69      30-25#
SLDEMP    5-70      31-5#
SLDFRE    5-71      30-40#
SLDNWR    5-73      30-35#
SLDWRT    5-72      30-30#
SLEHD     5-78#     34-12
SLKDOF    1-55      34-40     34-88
SLKDON    1-55      34-34     34-81
SLMXLN    1-203     34-60
SLOKED    5-80      5-81      5-82      34-80#
SLOLET    5-86      5-92      34-68#
SLONLT    5-87      5-93      34-74#
SLONOP    5-79      5-85      5-94      5-98      5-99      34-100#
SLONTT    5-96      34-53#
SLOOFF    5-88      34-40#
SLOON     5-89      5-90      34-23#
SLORT     5-83      5-91      34-87#
SLOTTY    5-95      34-47#
SLOUNI    5-84      34-94#
```

Cross reference table (CREF V05.05)

```
SLOWID   5-97      34-59#
SMRSIZ   1-143
SO$NO    1-75      12-43
SO$NVL   1-75      12-48
SO$OCT   1-75      12-56
SOPALC   1-66
SOPDAT   1-66
SOPTIM   1-66
SPACE1   1-49
SPACE2   1-179
SPACE3   1-179
SPACE5   1-180
SPACE6   1-183
SPACTV   1-187      1-188
SPCF     1-189
SPDTX1   1-178
SPFLK    1-189
SPFUL    1-189
SPGEMT   1-189
SPLACT   1-192
SPLCHN   1-92
SPLHD    1-186
SPLHLA   1-177     25-63
SPLPND   1-194
SPSNG    1-187      1-189
SPUBUF   1-69
SPWFM    1-187      1-188
SRTSIZ   1-143
SRTSMS   1-194
SRTTXT   1-197
SSRMAP   1-197
STCDOF   5-238     10-40#
STCDON   5-237     10-32#
STCHNB   4-9       10-4#
STLGAL   5-212     24-62#
STLGCL   5-214     24-15#
STLGCN   1-27       5-213     24-20#
STLGFL   5-215     25-4#
STLGHD   1-176      5-211#    24-9
STLGIN   5-216     24-50#
STLGNW   5-219     24-45#
STLGOT   5-217     24-56#
STLGWR   5-218     24-40#
STLOFL   5-225     26-21#
STLOHD   5-224#    26-10
STNOVR   1-34       4-60
STPASK   1-194
STPFLG   1-92      23-17*    23-25*
STRWRD   5-181      5-182    14-22#
STUCLF   5-53      14-61#
STUCLL   5-55      14-73#
STUCLM   5-54      14-67#
STUCLN   5-56      14-79#
STVRFY   1-34       4-59
SUBARG   1-184
SUBTXT   1-197
```

Cross reference table (CREF V05.05)

SUCS 1-144
SUM1 1-193
SUM2 1-193
SUM3 1-193
SUM4 1-193
SUM5 1-194
SUM6 1-194
SUM7 1-194
SUMS 1-144
SUPCOD 1-144
SWPTX 1-180
SXBPNT 1-69
SYASHD 1-59
SYHD1 1-179
SYHD2 1-179
SYINDX 1-151
SYNAME 1-152 28-31
SYPSWD 1-39 16-24
SYSAV 1-164
SYSDAT 1-141
SYTIMH 1-141
SYTIML 1-141
SYUNIT 1-151
TAB 2-8#
TALEMT 1-114
TBLOVF 1-173
TC132 6-21# 18-60
TC80 6-22# 18-66
TCNORM 6-20# 18-78
TCREV 6-19# 18-72
TECO 1-73
TK1SEC 1-143
TK1VAL 1-141
TM$AUT 1-57
TM$CDS 1-56
TM$CEN 1-56
TM$CL0 1-58
TM$CL1 1-58
TM$CL2 1-58
TM$CL3 1-58
TM$CL4 1-58
TM$CL5 1-58
TM$CL6 1-58
TM$CNG 1-56
TM$GBL 1-49
TM$HPE 1-56
TM$HPR 1-55
TM$IN1 1-38
TM$IN2 1-38
TM$LCL 1-49
TM$LPR 1-55
TM$MRS 1-32 13-24
TM$NAD 1-60
TM$NNR 1-38
TM$NSD 1-60
TM$PR1 1-55

Cross reference table (CREF V05.05)

```
TM$PR2    1-55
TM$PVA    1-44
TM$PVC    1-44
TM$RD1    1-49
TM$RD2    1-49
TM$SDN    1-60
TM$XBK    1-37
TMIDLH    1-71
TMIOH     1-71
TMIOWH    1-70
TMSWPH    1-71
TMSWTH    1-71
TMTOTH    1-70       1-193
TMTOTL    1-70       1-193
TMUSRH    1-70
TOTMMS    1-197
TOTON     1-92
TOTXT     1-176     21-11      22-11
TRGRET    1-144
TRMHD1    1-56
TRMHD2    1-56
TRMSTR    1-166
TSKST1    1-5#       1-26
TSR       1-156
TSXLN     1-144
TSXSIT    1-144
TSXSMS    1-198
TXTCL     1-91
UC$MDC    1-163
UC$NDC    1-163
UCHAN     1-112
UCIDEF    1-64      28-30
UCISPC    1-96      28-36      28-42
UCLBLK    1-162
UCLCMD    1-26
UCLDAT    1-162
UCLHD     4-57       5-52#
UCLNAM    1-117
UERSEV    1-134
UFORM     1-104
UFPTRP    1-119
UHIMEM    1-107
UKMNAM    1-85
UMSSMS    1-197
UMSYTP    1-89
UPTMMS    1-192
USPLCH    1-92
USRMMS    1-198
USRSTK    1-68
USTART    1-103
UTRPAD    1-67
VCORTM    1-138      4-11
VCSHNB    1-95      10-22*
VDBFLG    1-82      10-32
VHIPCT    1-136      4-18
VIMAGE    1-102
```

Cross reference table (CREF V05.05)

```
VINTIO    1-128     4-21
VLDSYS    1-86      7-56
VNUMDC    1-138     9-49
VOFFTM    1-205     4-29
VONTM     1-205     4-30
VPRIDF    1-42     22-15
VQUAN0    1-135     1-136     4-37
VQUAN1    1-136     4-38
VQUAN2    1-136     4-42
VQUAN3    1-135     1-136     4-43
VQUN1A    1-136     4-39
VQUN1B    1-128     4-40
VQUN1C    1-128     4-41
VSLEDT    1-203    34-23
VT100     1-147    18-44     34-27
VT10FL    1-149
VT10NO    1-149
VT200     1-146    18-44     34-27
VT2007    1-146
VT2008    1-146
VT20FL    1-150
VT20NO    1-150
VT52      1-147    18-44     34-27
VT52FL    1-148
VT52NO    1-126
VTMIN     1-205     4-54
VTMLOC    1-205     4-55
VTMOUT    1-205     4-56
VUSPHN    1-205     4-31
WEM      18-91     18-96#    18-100
WF132     5-123     6-14#     18-38     18-58
WF80      5-116     6-15#     18-64
WFDEL     5-114     5-118     6-11#     18-30
WFNORM    5-113     5-117     6-13#     18-76
WFREV     5-115     5-120     6-12#     18-70
WILDFL    1-72
WILDHD    4-61      5-24#
WINBIT    5-113     5-114     5-115     5-116     5-117     5-118     5-119     5-120     5-123     19-56#
WINCOL    5-112    19-4#
WINDEL    6-55#    18-32
WINFLG    6-7#     18-17*    18-26     19-56*
WINHD     5-111#   18-21
WINMAK    6-43#    18-15*    18-16*    18-40*    18-47     19-12*    19-40*    19-50*
WINMAP    6-50#    18-53
WINNOS    5-122    19-50#
WINSCR    5-121    19-23#
WLDNAM    2-12#
XAREA     1-165    11-28     11-30     11-41     12-75     12-96     12-97     13-5      14-87     14-90     14-97     14-100
         23-9     25-42     28-42     29-6      29-11     29-30     29-35     29-50     29-52
YESTXT    1-178
ZCLR      1-156
```

Cross reference table (CREF V05.05)

```
...CM0   25-28
...CM1   11-28   11-30   11-41   12-75   12-96   12-97   13-5    25-42   28-42   29-6    29-11   29-30
         29-35   29-50   29-52
...CM2   11-28   11-28   11-30   11-30   11-30   11-30   11-41   12-75   12-96   12-97   13-5    13-5
         13-5    13-5    14-87   14-90   14-90   14-97   14-100  14-100  23-9    23-9    25-42   25-42
         25-42   28-42   28-42   29-6    29-6    29-11   29-11   29-11   29-11   29-30   29-35   29-50
         29-50   29-52
...CM3    8-57   11-35   11-44   12-74   12-95   13-6    13-38   28-44   29-34
...CM5    9-28    9-30   10-16   10-18   11-28   11-30   11-41   12-75   12-96   12-97   13-5    14-87
         14-90   14-97   14-100  18-60   18-66   18-72   18-78   21-11   21-14   21-21   22-11   22-14
         23-9    25-28   25-42   28-42   29-6    29-11   29-30   29-35   29-50   29-52   31-21   34-34
         34-40   34-62   34-81   34-88
...CM6   14-87   14-90   14-97   14-100  23-9
...CM7   11-30   13-5    29-11
.CLOSE    1-20#  13-6    28-44
.CSIGE    1-18#
.CSISP    1-16#
.DATE     1-19#
.DSTAT    1-22#  25-28
.ENTER    1-21#  25-42
.EXIT     1-21#  29-39
.FPROT    1-22#
.GTIM     1-19#
.GTLIN    1-19#
.GVAL     1-22#  14-87   14-97
.HERR     1-22#
.LOOKU    1-20#  11-28   28-42   29-6    29-50
.PRINT    1-20#   9-28    9-30   10-16   10-18   18-60   18-66   18-72   18-78   21-11   21-14   21-21
         22-11   22-14   31-21   34-34   34-40   34-62   34-81   34-88
.PURGE    1-17#   8-57   11-35   11-44   12-74   12-95   13-38   29-34
.PVAL     1-22#  14-90   14-100  23-9
.READW    1-17#  11-30   29-11
.REOPE    1-18#  12-75   12-96   12-97   29-35
.SAVES    1-18#  11-41   29-30   29-52
.SERR     1-22#
.SPFUN    1-19#
.SRESE    1-16#
.TTOUT    1-16#
.TTYIN    1-17#
.TTYOU    1-17#
.WRITW    1-21#  13-5
CMDDEF    3-47#   4-9    4-10    4-11    4-12    4-13    4-14    4-15    4-16    4-17    4-18    4-19
          4-20    4-21    4-22    4-23    4-24    4-25    4-26    4-27    4-28    4-29    4-30    4-31
          4-32    4-33    4-34    4-35    4-36    4-37    4-38    4-39    4-40    4-41    4-42    4-43
          4-44    4-45    4-46    4-47    4-48    4-49    4-50    4-51    4-52    4-53    4-54    4-55
          4-56    4-57    4-58    4-59    4-60    4-61    4-62    4-63    5-5     5-6     5-7     5-8
          5-14    5-15    5-16    5-17    5-18    5-19    5-25    5-26    5-32    5-33    5-34    5-35
          5-36    5-37    5-38    5-39    5-45    5-46    5-47    5-53    5-54    5-55    5-56    5-62
          5-63    5-69    5-70    5-71    5-72    5-73    5-79    5-80    5-81    5-82    5-83    5-84
          5-85    5-86    5-87    5-88    5-89    5-90    5-91    5-92    5-93    5-94    5-95    5-96
          5-97    5-98    5-99    5-105   5-106   5-112   5-113   5-114   5-115   5-116   5-117   5-118
          5-119   5-120   5-121   5-122   5-123   5-129   5-130   5-131   5-132   5-133   5-134   5-135
          5-136   5-137   5-138   5-139   5-140   5-141   5-142   5-148   5-149   5-150   5-151   5-152
          5-153   5-154   5-155   5-156   5-157   5-158   5-159   5-160   5-161   5-167   5-168   5-174
          5-175   5-181   5-182   5-188   5-189   5-190   5-191   5-192   5-193   5-194   5-195   5-196
          5-197   5-198   5-199   5-200   5-201   5-202   5-203   5-204   5-205   5-206   5-212   5-213
```

```
          5-214     5-215     5-216     5-217     5-218     5-219     5-225     5-231     5-237     5-238
FABORT    3-15#     7-44      8-56      8-58      10-34     10-37     10-47     11-36     11-45     13-33     13-39     14-39
          14-92     16-19     18-11     18-46     18-92     22-18     25-44     25-80     26-25     26-57     27-26     27-33
          27-50     28-51     28-55     28-59     29-43     29-54     30-17
FERR      3-4#      15-38     20-41     20-43     21-8      21-18     22-8
FWARN     3-23#     13-16     13-24     34-25     34-29     34-59     34-94     34-108
TBLDEF    3-35#     4-8       5-4       5-13      5-24      5-31      5-44      5-52      5-61      5-68      5-78      5-104
          5-111     5-128     5-147     5-166     5-173     5-180     5-187     5-211     5-224     5-230     5-236
TBLEND    3-61#     4-64      5-9       5-20      5-27      5-40      5-48      5-57      5-64      5-74      5-100     5-107
          5-124     5-143     5-162     5-169     5-176     5-183     5-207     5-220     5-226     5-232     5-239
```



```
TTTTT  SSS  K   K  SSS  TTTTT  222
  T   S   S K  K  S   S   T   2   2
  T   S     K K   S       T       2
  T    SSS  KK     SSS    T      2
  T       S K K       S   T     2
  T   S   S K  K  S   S   T    2
  T    SSS  K   K  SSS    T   22222
```

```
TTTTTTTTTT    SSSSSS    KK      KK    SSSSSS    TTTTTTTTTT    222222
TTTTTTTTTT    SSSSSS    KK      KK    SSSSSS    TTTTTTTTTT    222222
    TT      SS      SS  KK    KK    SS      SS      TT      22      22
    TT      SS      SS  KK    KK    SS      SS      TT      22      22
    TT      SS          KK  KK      SS              TT              22
    TT      SS          KK  KK      SS              TT              22
    TT        SSSSSS    KKKK          SSSSSS        TT            22
    TT        SSSSSS    KKKK          SSSSSS        TT            22
    TT              SS  KK  KK              SS      TT          22
    TT              SS  KK  KK              SS      TT          22
    TT      SS      SS  KK    KK    SS      SS      TT        22
    TT      SS      SS  KK    KK    SS      SS      TT        22
    TT        SSSSSS    KK      KK    SSSSSS        TT        2222222222
    TT        SSSSSS    KK      KK    SSSSSS        TT        2222222222
```

```
[[[   000  55555       000   333    ]]]
[    0   0 5          0   0 3   3     ]
[    0  00 5          0  00     3     ]
[    0 0 0 5555       0 0 0   33      ]
[    00  0     5  ,,  00  0     3     ]
[    0   0 5   5  ,,  0   0 3   3     ]
[[[   000   555  ,,    000   333    ]]]
```

```
L     IIIII N   N EEEEE              1     4
L       I   NN  N E                 11    44
L       I   N N N E        =====   1 1   4 4
L       I   N  NN EEEE               1   44444
L       I   N   N E        =====     1      4
```