6.31

```
                                .TITLE  TSGEN -- System Generation Parameters
                                .IDENT  /V6.31/
000000                          .CSECT  TSGEN
                                .ENABL  LC
                                .DSABL  GBL
                                .NLIST  CND
                        ;-------------------------------------------------------------------------------
                        ;  TSGEN version 6.31
                        ;
                        ;  This module contains the the definitions of system parameters
                        ;  that define the characteristics of the TSX-Plus system
                        ;  being generated.
                        ;
                        ;  Written by Phil Sherrod.
                        ;
                        ;  Copyright (c) 1978,1979,1980,1981,1982,1983,1984,1985,1986,1987,1988.
                        ;  S&H Computer Systems, Inc.
                        ;  1027 17th. Avenue South
                        ;  Nashville, Tennessee  U.S.A.
                        ;  (615) 327-3670
                        ;
                        ;  This software is furnished under a license for use only on a
                        ;  single computer system and may be copied only with the inclusion
                        ;  of the above copyright notice.
                        ;  This software, or any copies thereof, may not be provided
                        ;  or otherwise made available to any other person except for
                        ;  use on such system and to one who agrees to these license terms.
                        ;  Title to and ownership of the software shall at all times remain
                        ;  with S&H Computer Systems, Inc.
                        ;  This software is the valuable property of S&H Computer Systems, Inc.
                        ;  All rights reserved.
                        ;
                        ;  S&H will seek legal redress for any unauthorized use of this product.
                        ;
                        ;
                        ;  Set FULLST to 1 for a full assembly listing.
                        ;  Set FULLST to 0 for a normal short listing.
                        ;
000001                  FULLST  =       1
                        ;
000000                  TSGEN:
                        ;
                        ;  Global definitions
                        ;
                                .GLOBL  MXTTCT,HANDSK,HANRCB,HANRCO
                                .GLOBL  IOHANQ,GETUMR,FREUMR,SYPSWD,SYPSPR,TSXVRS
                                .GLOBL  TSGEN,RDBSEC,DTYPE,NAMSEC,LNMTOP,SASECT,SMONHD
                                .GLOBL  NPL,NSL,NLIN2,LSTHL,TNHL,NIOL,FSTIOL
                                .GLOBL  NLINES,NLCHN,$MEMSZ,MEMPTR,PHYMEM,CFSTS,CFABLV
                                .GLOBL  VNGR,SHRRCB,SHRRCN,SCPFHD,VNUIP,INSTBL,INSTBN
                                .GLOBL  SPLND,SPLNF,SPLNB,VU$CL,UCLNAM,MIOBHD,DETCBS
                                .GLOBL  SPLBHD,SFCBFH,KMNTOP,KMNHI,NESB,FASTIN
                                .GLOBL  SDCB,SDCBND,SPLDEV,SPLDVN,UKMNAM,IOHLTM
                                .GLOBL  NFRESB,SPLBLK,NSPLDV,VSWPSL,CLEOFS,VMXWIN
                                .GLOBL  INVEC,RSR,RBR,TSR,TTCSCH,VSLEDT,FRKINI,FRKGEN
                                .GLOBL  TBR,INRECV,OTRECV,LSTPL,LUNAME,LSTIOL,NUMFRK,NXIVMH
                                .GLOBL  QUAN1,QUAN2,LSTMX,R$TTOP,R$INST,LSTLIN,R$CFST
```

```
                                .GLOBL  INMXV,OTMXV,PVSPBL,LMXLN,LMXPRM,NDVRCB
                                .GLOBL  OTRAS7,LNMAP,LNPRIM,LPRI,VLDSYS,NSPLFL,NSPLBL
                                .GLOBL  CLVERS,CLORSZ,VMXMON,MONFQH,SPSTAT,LDVERS
                                .GLOBL  NCSILO,NCXOFF,NCXON,CSHALC,VUXIFL
                                .GLOBL  LSTSL,FSTSL,CTRLTT,VINABT,IOABFL
                                .GLOBL  MAXSEC,VEDIT,WILDFL,VPRIVR,CLXTRA,CLTOTL
                                .GLOBL  SWDBLK,TMIOL,TMIOH,TK1CNT,TK1SEC,NSCP
                                .GLOBL  TMSWPL,TMSWPH,TMIOWL,TMIOWH,VCSHNB
                                .GLOBL  DCTOTU,DCTRD,DCCRD,DCTWR,DCCWR
                                .GLOBL  SFCB,SFCBND,QUAN1A,LOGCHN,QUAN1B
                                .GLOBL  TSXSIT,CFCHAN,NUCHN,MVSIZ,VQUN1C,VNRFLG
                                .GLOBL  VDISPC,VDOSPC,VDMKTP,VSYDMP,VDMTCR,MODDAT
                                .GLOBL  VNCSLO,VNCXOF,VNCXON
                                .GLOBL  CMFSEC,CMFTOP,VKEYMX,VSCHED,VDSKBU,MODTIM
                                .GLOBL  VTSLCH,MXSPAC,PMSIZE,VVLSCH,VQUAN0,VQUAN3,VVPWCH
                                .GLOBL  MXRBUF,MXRING,MXBRK,MH$SCR,MH$RCR,MH$CAR
                                .GLOBL  MH$BCR,MH$BAR,MH$SSR,DHBFSZ,VCXTRM,VCXCTL
                                .GLOBL  VH$CSR,VH$DBR,VH$LPR,VH$LSR,RUNCHN
                                .GLOBL  VH$LCR,VH$BA1,VH$BA2,VH$BCR,CDX$PI
                                .GLOBL  CDX$DL,CDX$DZ,CDX$DH,CDX$VH,CDX$PC,CDX$PP
                                .GLOBL  VMSCHR,VMIOSZ,RPRCSR,RPRVEC,DWTYPE,CDX$QP
                                .GLOBL  VMAXMC,VMXMSG,MSGBAS,USRBAS,VMXMRB,WINBAS
                                .GLOBL  NUMDCD,VNUMDC,DCAGE,NUMCDB,VMIOBF
                                .GLOBL  LOKBAS,LOKMEM,TIOBAS,LOKCSH
                                .GLOBL  VQUAN1,VQUN1A,VQUN1B,VQUAN2,VCORTM
                                .GLOBL  MAPPAR,RDB,RDBEND,NUMRDB,MIODBG
                                .GLOBL  CSHDEV,CSHDVN,VMXCSH,VNFCSH,CSHHD
                                .GLOBL  MAXALC,ALCTBL,ALCEND,VOFFTM
                                .GLOBL  VTMOUT,SNDBX,USPLCH,UBUSMP,MEM256
                                .GLOBL  FREFRK,BOTDEV,BOTUNI,BOTCSR,MIOFLG
                                .GLOBL  AUTHAN,AHEND,SNBUX,UMSYTP,VINTIO
                                .GLOBL  VMXSF,VMXSFC,VMLBLK,RSFBLK,VPLAS,SEGCHN
                                .GLOBL  SDCBSZ,SDBULS,SPLANM,MIONWB,INDFIL
                                .GLOBL  NDL,LSTDL,FSTDL,VSWPFL,MAPUSR,R$MFMV
                                .GLOBL  CORUSR,LINNUM,MUXNUM,NUMON,PVON,TOTON
                                .GLOBL  STPFLG,MINTIM,RTVECT,VHIPCT,FRKDLY
                                .GLOBL  SYSCHN,SYCH0,SYCH1,SYCH2,SYCH3,SYCH4,SYCH5
                                .GLOBL  SYCH6,SYCH7,SYCH10,SYCH11,SYCH12,SYCH13
                                .GLOBL  SYCH14,SYCH15,SYCH16,SYCH17,SYCH20
                                .GLOBL  BLKEY,CHKEY,SYSDAT,DFLG,QCOMP,VPRILO,VPRIHI
                                .GLOBL  SPUSR,SYUNIT,SYSVER,SYSUPD,CONFIG,MAXBLK,VPRIDF
                                .GLOBL  PNAME,HANSIZ,HANENT,DVSTAT,PROSLT,VIDCSR
                                .GLOBL  DEVSIZ,MAXDEV,NUMDEV,SYTIML,SYTIMH,SPLCHN,HANIOC
                                .GLOBL  SWPCHN,NUMIOQ,NUMSYQ,FREIOQ,NMFREQ,INTSSZ
                                .GLOBL  MONVEC,TK1VAL,VHIMEM,CCLSAV,INDSAV
                                .GLOBL  INDDBL,INDTSV,INDDBS,R$INTC
                                .GLOBL  SYINDX,CONFG2,SYSGEN,MAXGVL,LDDEVX,CLDEVX,C1DEVX
                                .GLOBL  TMTOTL,TMTOTH,TMUSRL,TMUSRH,SYNAME
                                .GLOBL  TMSWTL,TMSWTH,TMIDLL,TMIDLH,NUMCCB
                                .GLOBL  VMXFIL,VECBAS,VDFMEM,SNMSHD,MXJADR
                                .GLOBL  NMSNMB,NMUMB,CORTIM,RF$WRT,GENTOP
                                .GLOBL  BASMAP,LOMAP,HIMAP,FREPGS
                                .GLOBL  JCXPGS,MXJMEM,DFJMEM,TK5VAL,TK3SVL
                                .GLOBL  R$CHN,R$DATE,R$UBAS,R$JOB,R$CH17
                                .GLOBL  R$XCHN,EXTCHN,MVWDS,TTOPTS,MAPSIZ
                                .GLOBL  DVFLAG,VBUSTP,QBUS,UNIBUS
                                .GLOBL  VUCLMC,UCLDAT,UCLBLK,VUCLOR,SR3FLG
```

```
                        .GLOBL  SMRSIZ, SRTSIZ, CSHSIZ, MIOWHD, MIOSYQ
                        .GLOBL  PROFLG, MPARFL, PIDPTR
                        .GLOBL  PROBRK, SPOLID, VDBFLG, PROODC
                        .GLOBL  HANPAR, MHNSIZ, KUSECK, USWPCH, R$SWPC
                        .GLOBL  CA$BLK, CA$DVU, CA$HBL, CA$HFL, CA$HSH, CA$UBL, CA$UFL, CA$WCT
                        .GLOBL  CSHBFP, CSHLRU, CSHMRU, CSHFHD, CCBHD
                        .GLOBL  CASTBR, CASCBR, CASTBW, CASCUP, CASTRO, CASTWO
                        .GLOBL  CSHIO, CSHINI, CSHCLN, CSHFIN, CSHBAS, CSHVEC
                        .GLOBL  LOKVEC, LOKINI, DOOPAP, DOCOPN, SFSVST, SFRSST, DORLK
                        .GLOBL  DOTLK, DOCULK, DOULK1, DOSFCK, SFCLS, SFWRIT, CLSCDB
                        .GLOBL  DCRD1, DCRD2, LOKVEC, CLKVEC, VUSPHN, ABRTOV
                ;
                ;  Global references
                ;
                        .GLOBL  ININT, DLINT, LINNUM, PROITP, FNDHRB
                        .GLOBL  SD$HLD, MUXNUM, RMNBAS, SETERR
                        .GLOBL  IOFIN, INTEN, PSW, SYNCH
                        .GLOBL  LA36, LA120, VT52, VT100, ADM3A, VT200
                        .GLOBL  DIABLO, QUME, HAZEL, TSDEFS, EXTP1, HANXMR, BLKMV, CVTPHY
                        .GLOBL  GETRTQ, QFREE, IOSTRT, FRKGET, FORKQ, QIO, QCOMPL
                        .GLOBL  SCHED, DSKBUF, IOQSIZ
```

```
                        ;-------------------------------------------------------------------------------
                        ;  Internal parameters
                        ;
001167                  TSXVRS  =       631.            ;TSX-Plus version number
000017                  MAXDEV  =       15.             ;Max number of devices that can be supported
000036                  NUMFRK  =       30.             ;# Fork request blocks
000000                  NXIVMH  =       0.              ;# extra interrupt vectors for mapped handlers
000120                  DETCBS  =       80.             ;# characters for detached job startup cmnd.
000004                  FRKGEN  =       4.              ;# Fork blocks in TSGEN
000004                  NMSNMB  =       4.              ;# System message buffers
000002                  NMSYMB  =       2.              ;# message buffer reserved for system use
000024                  NUCHN   =       20.             ;# I/O channels user may use
000010                  MAXMUX  =       8.              ;Max # of DZ11's, DH11's, and DHV11's
000040                  DHBFSZ  =       32.             ;# bytes for DH11 and DHV11 DMA output buffers
000012                  TTCSCH  =       10.             ;# characters printed per scheduler check
000050                  NUMIOQ  =       40.             ;# I/O queue elements
000017                  NUMSYQ  =       15.             ;# I/O queue elements reserved for system I/O
000017                  NUMCCB  =       15.             ;# data cache control blocks
000010                  NSCP    =       8.              ;# swapper command packets
000714                  INTSSZ  =       460.            ;# bytes for system interrupt stack
000006                  MIONWB  =       6.              ;Number of mapped I/O wait queue elements
000000                  MIODBG  =       0.              ;1=Force I/O mapping (debugging use only)
000000                  MPARFL  =       0.              ;1==>Enable mem parity traps, 0==>disable.
000000                  PROBRK  =       0.              ;1==>Enable ODT break on PRO printer port
000003                  PROODC  =       3.              ;Clock ticks per PI driver call
000000                  FASTIN  =       0.              ;1==>Clock driven input character processing
000100                  CLKVEC  =       100             ;Clock interrupt vector
000006                  DCAGE   =       6.              ;Shared file data cache ageing factor
000001                  KUSECK  =       1.              ;0==>Don't check for usage on INIT & SQUEEZE.
000024                  IOHLTM  =       20.             ;# 0.1 secs I/O can be held for job swapping
000007                  CLEOFS  =       7.              ;Max num of chars in CL ENDSTRING parameter
000002                  NDRTDF  =       2.              ;Number of dummy shared run-time definitions
000000                  $PRIV   =       0               ;Obsolete line-def privilege flag
000020                  $NOVLN  =       20              ;Obsolete no-virtual-line privilege flag
                        ;
                        ;-------------------------------------------------------------------------------
                        ;  Fork priority values.
                        ;  Unlike RT-11, TSX-Plus assigns priority values to its fork requests
                        ;  and allows higher priority fork requests to interrupt lower priority ones.
                        ;  The priority values range from 1 to 127.  The higher the numerical value,
                        ;  the higher the priority.
                        ;
                                .GLOBL  FP$RT,FP$CKT,FP$DEF,FP$IOS,FP$IOF,FP$IOA,FP$MOV
                                .GLOBL  FP$CDI,FP$CDO,FP$CK1,FP$MAX,FP$FLG,FP$PIO
                        ;
100000                  FP$FLG  =       100000          ;Flag saying this is a priority value
100177                  FP$MAX  =       FP$FLG+127.     ;Max legal fork priority
100144                  FP$RT   =       FP$FLG+100.     ;Real-time interrupts
100106                  FP$CKT  =       FP$FLG+70.      ;50/60 Hz clock interrupt processing
100074                  FP$CDI  =       FP$FLG+60.      ;Terminal character input processing
100067                  FP$CDO  =       FP$FLG+55.      ;Terminal character output processing
100062                  FP$DEF  =       FP$FLG+50.      ;Default fork priority
100062                  FP$IOF  =       FP$FLG+50.      ;I/O finish
100062                  FP$IOA  =       FP$FLG+50.      ;I/O abort entry
100062                  FP$PIO  =       FP$FLG+50.      ;PI output interrupt processing
100036                  FP$CK1  =       FP$FLG+30.      ;0.1 second clock processing
                        ;  The following fork priorities are entered from a non-interrupt state.
```

```
                                FP$IOS  =       FP$FLG+12.      ;I/O initiation
                                FP$MOV  =       FP$FLG+10.      ;Move data to/from cache buffer
                                ;
                                ;-------------------------------------------------------------------------------
                                ;  Completion routine class priorities.
                                ;  A completion routine with a higher (numerically larger) class priority
                                ;  is allowed to interrupt a lower class priority completion routine.
                                ;
                                        .GLOBL  CP$STD,CP$RT,CP$SYN
                                ;
                                CP$STD  =       1               ;Standard -- I/O completion, .TIMIO, etc.
                                CP$RT   =       2               ;Real-time completion routine
                                CP$SYN  =       3               ;.SYNCH completion routine
                                ;
                                ;  Type codes used to identify communication device controllers
                                ;
                                CDX$DL  =       0               ;DL11
                                CDX$DZ  =       2               ;DZ11
                                CDX$DH  =       4               ;DH11
                                CDX$VH  =       6               ;DHV11
                                CDX$PI  =       10              ;Console terminal on Professional
                                CDX$PC  =       12              ;Communications port on Professional
                                CDX$PP  =       14              ;Printer port on Professional
                                CDX$QP  =       16              ;4 line Multiplexer on Professional
                                ;-------------------------------------------------------------------------------
                                CFCHAN  =       NUCHN           ;Channel to use for command file input
                                LOGCHN  =       NUCHN+1         ;Channel for log file
                                USPLCH  =       NUCHN+2         ;Channel to use to write to spool file
                                RUNCHN  =       NUCHN+3         ;Channel to use when loading a SAV file
                                USWPCH  =       NUCHN+4         ;Channel to use to access swap file
                                NLCHN   =       NUCHN+5         ;Total # channels allocated per job
                                LSTMX   =       0               ;Index to last mux
                                CURMX   =       0               ;Current mux #
                                NUMRDB  =       0               ;Count number of shared run-times declared
                                CURCDX  =       CDX$DL          ;Comm device type for current line
                                DHUSE   =       0               ;Set to 1 if DH11 or DHV11 support needed
```

```
                                ;-----------------------------------------------------------------------
                                ;  Monitor fixed-offset value vector
                                ;
                                ;  Table of addresses of TSX-Plus routines.  The pointer to this vector is
                                ;  stored at simulated RMON offset -2.
                                ;
                                        ;Negative offsets from TSXVEC reserved to users
000000  002260'          TSXVEC: .WORD   NUMDEV          ;  0 Ptr to word with # devices in system
000002  000576'                  .WORD   HANENT          ;  2 Vector with handler entry points
000004  001026'                  .WORD   HANPAR          ;  4 64-byte phys mem base of mapped handlers
000006  000000G                  .WORD   GETRTQ          ;  6 Routine to get a free I/O queue element
000010  000000G                  .WORD   QFREE           ; 10 Routine to free an I/O queue element
000012  000000G                  .WORD   QIO             ; 12 Routine to queue an I/O request
000014  000000G                  .WORD   IOSTRT          ; 14 Routine to requeue an I/O request
000016  000000G                  .WORD   QCOMPL          ; 16 Routine to queue a completion request
000020  000000G                  .WORD   FRKGET          ; 20 Routine to get a fork request block
000022  000000G                  .WORD   FORKQ           ; 22 Routine to queue a fork request
000024  000000G                  .WORD   IOHANQ          ; 24 Place I/O queue element on handler list
000026  000000G                  .WORD   GETUMR          ; 26 Allocate a Unibus map register
000030  000000G                  .WORD   FREUMR          ; 30 Free a Unibus map register
000032  001167                   .WORD   TSXVRS          ; 32 TSX-Plus version number
000034  000000G                  .WORD   IOQSIZ          ; 34 Size of an I/O queue element (bytes)
                                ;
                                ;  Macro to reserve I/O channel space.
                                ;
                                        .MACRO  CHNRES
                                        .WORD   0,0,0,0,0
                                        .ENDM   CHNRES
                                ;-----------------------------------------------------------------------
                                ;  Fixed-offset vector
                                ;
                                ;  The following vector of addresses and values corresponds to the fixed
                                ;  offset cells in RT-11 RMON.  These cells are mapped into user
                                ;  address space through PAR7 (160000 - 177777).
                                ;
000036  000000'          VECBAS: .WORD   TSXVEC          ;  -2 Pointer to vector of TSX addresses
000040  000167  000000G  MONVEC: JMP     INTEN           ;   0 Handler interrupt entry point
                                ;
                                ;  System channel space
                                ;
000044                  SYSCHN:
000044                  SYCH0:  CHNRES
000056                  SYCH1:  CHNRES
000070                  SYCH2:  CHNRES
000102                  SYCH3:  CHNRES
000114                  SYCH4:  CHNRES
000126                  SYCH5:  CHNRES
000140                  SYCH6:  CHNRES
000152                  SYCH7:  CHNRES
000164                  SYCH10: CHNRES
000176                  SYCH11: CHNRES
000210                  SYCH12: CHNRES
000222                  SYCH13: CHNRES
000234                  SYCH14: CHNRES
000246                  SYCH15: CHNRES
000260                  SYCH16: CHNRES
```

```
                                SYCH17: CHNRES
                                SYCH20: CHNRES
                                ;
000316  000000                  BLKEY:  .WORD   0               ;256 - # of directory block that is in core
000320  000000                  CHKEY:  .WORD   0               ;260 - # of device whose dir block is in core
000322  000000                  SYSDAT: .WORD   0               ;262 - System date word
000324  000000                  DFLG:   .WORD   0               ;264 - Directory op is in progress
                                ;
                                ;  The following cells are documented for access by .GVAL
                                ;
000326  000100                  USROFF: .WORD   HIMEM           ;266 - Base of USR
000330  000000G                 QCOMP:  .WORD   IOFIN           ;270 - I/O completion handler entry point
000332  000000                  SPUSR:  .WORD   0               ;272 - USR error cell
000334  000000                  SYUNIT: .WORD   0               ;274 - Unit # of SY device
000336     004                  SYSVER: .BYTE   4               ;276 - System version number
000337     000                  SYSUPD: .BYTE   0               ;277 - Release #
000340  000000                  CONFIG: .WORD   0               ;300 - System configuration word
000342                                  .BLKW   5.              ;302 - 313 (unused)
000354  001750                  MAXBLK: .WORD   MAXFIL          ;314 - Largest output file size
                                ;  Word 316 in TSX-Plus is reserved for specific use.  It must
                                ;  be initialized to zero and not used by the operating system.
                                ;  See the note in TSDEFS for more specific information.
000356  000000  000000                  .WORD   0,0             ;316 - 321 unused
000362  000000                  CORUSR: .WORD   0               ;322 - Current job number
000364  000000G                         .WORD   SYNCH           ;324 - Address of .SYNCH request routine
000366                                  .BLKW   13.             ;326 - 357  unused
000420  000445                          BR      MTPS            ;360 - Move to PS routine
000422  000432                          BR      MFPS            ;362 - Move from PS routine
000424  000000                  SYINDX: .WORD   0               ;364 - Device number of system device
000426  000000                  CFSTS:  .WORD   0               ;366 - Command file status flags
000430  000000                  CONFG2: .WORD   0               ;370 - Extended configuration word
000432  000000                  SYSGEN: .WORD   0               ;372 - System generation options
000434  000002                          .WORD   2               ;374 - Size of USR
000436     014                  CFABLV: .BYTE   14              ;376 - Error abort severity level
000437     003                          .BYTE   3               ;377 - Max @file nesting level
000440  000000                  EMTRTN: .WORD   0               ;400 - EMT return point
000442  000000                  FRKADR: .WORD   0               ;402 - Fork routine
000444  000500                  PNPTR:  .WORD   PNAME-MONVEC    ;404 - Offset to permanent dev name table
000446  071677  142615          MONAME: .RAD50  /RT11XM/        ;406 - 410 - System name
000452  000000                  HSUFFX: .WORD   0               ;412
000454  000000                  SPSTAT: .WORD   0               ;414 - Spooler status flags
000456     000                          .BYTE   0               ;416 - Error byte for IND
000457     000                  INSTA:  .BYTE   0               ;417 - IND status byte
000460  000000                  $MEMSZ: .WORD   0               ;420 - Total 32-word mem blocks avail
000462  000000                          .WORD   0
000464  000442G                 $TCFIG: .WORD   TTOPTS+RMNBAS   ;424 - Address of TT config word
000466  000444G                 $INDDV: .WORD   INDOFF+RMNBAS   ;426 - Pointer to IND device name word
000470  001300                  MEMPTR: .WORD   HNMEPT-MONVEC   ;430 - Offset to memory control blocks
000472  001330'                 P1EXT:  .WORD   P1XPTR          ;432 - Kernel PAR1 routine
000474  000000                  RPRCSR: .WORD   0               ;434 - Get CSR address of PRO devices
000476  000000                  RPRVEC: .WORD   0               ;436 - Get vector address of PRO devices
000500  000000                  DWTYPE: .WORD   0               ;440 - Type of DW disk
                                ;  Dummy cell corresponding to cell in RT-11 with TT option flags.
        000442                  TTOPTS  =       .-MONVEC        ;Offset to TTOPTS cell
000502  000000                  TTOP:   .WORD   0               ;TTOPTS cell
                                ;  Cell with name of IND RUN device
        000444                  INDOFF  =       .-MONVEC        ;Offset to INDDEV cell
```

```
    115 000504      123      131      060  INDDEV: .ASCII  /SY0:/          ;Default device from which IND is run
        000507      072
    116                                      ;-------------------------------------------------------------------------
    117                                      ;  MFPS is called to return on the stack the contents of the low-order
    118                                      ;  byte of the processor status word.
    119                                      ;  Note: This only works when used within handlers.  If a .MFPS macro
    120                                      ;        is used in a TSX-Plus user job, a value of zero is returned.
    121                                      ;
    122                                      ;  Outputs:
    123                                      ;   Processor status word is on top of stack.
    124                                      ;
    125 000510  005046                       MFPS:    CLR     -(SP)
    126 000512  013716  0000000G             MFPMOV:  MOV     @#PSW,(SP)      ;Get the psw (** patched during job init **)
    127 000516  016646  000002               MOV     2(SP),-(SP)     ;Now push return address on top
    128 000522  016666  000002  000004       MOV     2(SP),4(SP)     ;Move down PS value
    129 000530  012616                       MOV     (SP)+,(SP)      ;Move down return address
    130 000532  000207                       RETURN
    131                                      ;
    132                                      ;  MTPS is called to set the value of the low-order byte of the
    133                                      ;  processor status word.
    134                                      ;
    135                                      ;  Inputs:
    136                                      ;   Value to be moved to psw is on top of stack before call.
    137                                      ;
    138                                      ;  Outputs:
    139                                      ;   Value is moved to psw and popped from the stack.
    140                                      ;
    141 000534  000006                       MTPS:    RTT                    ;PC&PS are on stack, let RTT set PS and return
    142                                      ;
    143                                      ;-------------------------------------------------------------------------
    144                                      ;  Device and handler information tables (Do not change the order).
    145                                      ;
    146                                      ;  *** VM depends on PHYMEM being 1 word below PNAME ***
    147                                      ;
    148 000536  000000                       PHYMEM: .WORD   0               ;*** Store actual physical memory size ***
    149 000540  000017                       PNAME:  .REPT   MAXDEV          ;Table of permanent device names (Rad50)
    150                                              .WORD   0
    151                                              .ENDR
    152 000576  000017                       HANENT: .REPT   MAXDEV          ;Handler entry point
    153                                              .WORD   0
    154                                              .ENDR
    155 000634  177777                               .WORD   -1              ;Flag to mark end of HANENT table
    156 000636  000017                       DVSTAT: .REPT   MAXDEV          ;Device status flags
    157                                              .WORD   0
    158                                              .ENDR
    159 000674  000017                       HANDSK: .REPT   MAXDEV          ;Location of handler on the disk
    160                                              .WORD   0
    161                                              .ENDR
    162 000732  000017                       HANSIZ: .REPT   MAXDEV          ;Size of device handler
    163                                              .WORD   0
    164                                              .ENDR
    165 000770  000017                       DEVSIZ: .REPT   MAXDEV          ;# 256-word blocks on device
    166                                              .WORD   0
    167                                              .ENDR
    168 001026  000017                       HANPAR: .REPT   MAXDEV          ;64-byte base block for handler if mapped
    169                                              .WORD   0
    170                                              .ENDR
```

```
                                HANIOC: .REPT   MAXDEV          ;# uncompleted I/O requests for handler
                                        .WORD   0
                                        .ENDR
                                DVFLAG: .REPT   MAXDEV          ;Table of device characteristics
                                        .WORD   0
                                        .ENDR
        001120                  MAXGVL  =       .-MONVEC        ;Max offset allowed with .GVAL
                                ;-------------------------------------------------------------------------
                                ;
                                ;  Reserve space for extended channel space
                                ;
001160                          EXTCHN:
        000010                          .REPT   <NLCHN-17.>
                                        CHNRES
                                        .ENDR
                                ;
                                ;  Reserve space for channel used to access INDTMP file
                                ;
001300                          INDTSV: CHNRES                  ;Channel used for I/O to INDTMP file
                                ;
                                ;  Address of channel block for swap file access
                                ;
        001266'                 SWPCHN  =       EXTCHN+<10.*<USWPCH-17.>>
                                ;
                                ;  End of MONVEC pointer table area.
                                ;
        001254                  MVSIZ   =       .-VECBAS
        000526                  MVWDS   =       MVSIZ/2         ;# words in mon vector table
                                ;
                                ;  Define offsets into monitor vector area
                                ;
        000006                  R$CHN   =       SYSCHN-VECBAS   ;Start of channel space
        000234                  R$CH17  =       SYCH17-VECBAS   ;Offset to channel # 17
        001122                  R$XCHN  =       EXTCHN-VECBAS   ;Offset to extended channel space
        000264                  R$DATE  =       SYSDAT-VECBAS   ;Offset to date word
        000324                  R$JOB   =       CORUSR-VECBAS   ;Offset to job number cell
        000270                  R$UBAS  =       USROFF-VECBAS   ;Offset to usr base address cell
        000444                  R$TTOP  =       TTOP-VECBAS     ;Offset to TT option word
        000421                  R$INST  =       INSTA-VECBAS    ;Offset to IND status byte
        000370                  R$CFST  =       CFSTS-VECBAS    ;Offset to command file status word
        001242                  R$INTC  =       INDTSV-VECBAS   ;Offset to INDTMP channel block
        001230                  R$SWPC  =       SWPCHN-VECBAS   ;Offset to USWPCH channel block
        000454                  R$MFMV  =       MFPMOV-VECBAS   ;Offset to MFPMOV instruction
                                ;
                                ;-------------------------------------------------------------------------
                                ;  Vector of entry points to handler support routines
                                ;  (Do not alter the order)
                                ;
001312  000167  000000G                 JMP     CVTPHY          ;Routine to convert virtual to physical addr
001316  000167  000000G                 JMP     FNDHRB          ;Routine to search for RCB for handler
001322  000167  000000G                 JMP     HANXMR          ;Routine to allocate XM region for handler
001326  000402                          BR      BMJMP           ;Routine to do block move
001330  000167  000000G         P1XPTR: JMP     EXTP1           ;Routine to execute mapped code
001334  000167  000000G         BMJMP:  JMP     BLKMV
                                ;
                                ;-------------------------------------------------------------------------
                                ;  Memory allocation information for handlers
```

```
228                                     ;
229 001340  000000                      HNMEPT: .WORD   0
230 001342  000000                              .WORD   0
231 001344  000000                      HANRCO: .WORD   0
```

```
                                ;-----------------------------------------------------------------------------
                                ;  Misc data cells
                                ;
001346  000002                VQUANO: .WORD   QUANO
001350  000024                VQUAN1: .WORD   QUAN1
001352  000002                VQUN1A: .WORD   QUAN1A
001354  000002                VQUN1B: .WORD   QUAN1B
001356  000001                VQUN1C: .WORD   QUAN1C
001360  000012                VQUAN2: .WORD   QUAN2
001362  000024                VQUAN3: .WORD   QUAN3
001364  000002                VCORTM: .WORD   CORTIM
001366  000050                VHIPCT: .WORD   HIPRCT
001370  000036                VINTIO: .WORD   INTIOC
001372  000036                VMXSF:  .WORD   MAXSF
001374  000036                VMXSFC: .WORD   MAXSFC
001376  000000                VNUMDC: .WORD   NUMDC
001400  000003                VMLBLK: .WORD   MXLBLK
001402  000005                VUCLMC: .WORD   UCLMNC
001404  001750                VMXFIL: .WORD   MAXFIL
001406  000050                VNFCSH: .WORD   NMFCSH
001410  000005                VMXMON: .WORD   MAXMON
001412  000036                VTMOUT: .WORD   TIMOUT
001414  000074                VOFFTM: .WORD   OFFTIM
001416  000003                VMAXMC: .WORD   MAXMC
001420  000310                VMSCHR: .WORD   MSCHRS
001422  000003                VMXMSG: .WORD   MAXMSG
001424  000012                VMXMRB: .WORD   MAXMRB
001426  000100                VHIMEM: .WORD   HIMEM
001430  000070                VDFMEM: .WORD   DFLMEM
001432  000007                VSWPSL: .WORD   SWPSLT          ;# of job slots in swap file
001434  000000                VPLAS:  .WORD   SEGBLK          ;# blocks for PLAS swap file
001436  000014                VNGR:   .WORD   NGR             ;Number of global PLAS regions
001440  000004                NDVRCB: .WORD   DEVXMR          ;Number of PLAS regions for device handlers
001442  000012                VMXWIN: .WORD   MAXWIN          ;Maximum number of display windows
001444  000007                VKEYMX: .WORD   KEYMAX          ;Maximum # user-defined keys
001446  000004                VNUIP:  .WORD   NUIP            ;Number of user programs that may be INSTALLed
001450  000000                VCSHNB: .WORD   CACHE           ;# blocks in use for generalized data cache
001452  000000                CSHALC: .WORD   CACHE           ;# blocks allocated for generalized data cache
001454  040150                VNRFLG: .WORD   NRMFLG          ;Default time-sharing line flags
001456  000000G               VSCHED: .WORD   SCHED           ;An entry point in TSEXEC
001460  000000G               VDSKBU: .WORD   DSKBUF          ;A global from TSINIT
001462  000040                VNCSLO: .WORD   NCSILO          ;Default #bytes for TT and CL silos
001464     014                VNCXOF: .BYTE   NCXOFF          ;Default XOFF when only this many free
001465     004                VNCXON: .BYTE   NCXON           ;Default XON when this many remain
001466  000144                VDISPC: .WORD   DINSPC          ;Default line input buffer size
001470  000360                VDOSPC: .WORD   DOTSPC          ;Default line output buffer size
001472  000000                SYTIMH: .WORD   0               ;High-order system time word
001474  000000                SYTIML: .WORD   0               ;Low-order system time word
001476  000000                TK1SEC: .WORD   0               ;# clock ticks per second
001500  000000                TK1VAL: .WORD   0               ;# clock ticks per 0.1 second
001502  000000                TK1CNT: .WORD   0
001504  000000                TK5VAL: .WORD   0               ;# clock ticks per 0.5 seconds
001506  000000                TK3SVL: .WORD   0               ;# clock ticsk per 3 seconds
001510  000000                TSXSIT: .WORD   0
001512  000000                FRKDLY: .WORD   0               ;Max clock ticks a fork request was delayed
001514  000000                CTRLTT: .WORD   0               ;# of operator's console
001516  000000                MINTIM: .WORD   0               ;Number of minutes of system up-time
```

```
 58 001520                          SEGCHN: .BLKW   5               ;Channel block used for PLAS region swapping
 59 001532  000000                  KMNTOP: .WORD   0               ;Abs address of top of TSKMON
 60 001534  000000                  KMNHI:  .WORD   0               ;KMNTOP-KMNBAS
 61 001536                          CCLSAV: .BLKW   5               ;Savestatus for CCL.SAV file info
 62 001550                          INDSAV: .BLKW   5               ;Savestatus for IND.SAV file info
 63 001562  000000                  INDDBL: .WORD   0               ;Lowest block in IND.SAV file of data segmemt
 64 001564  000000                  INDDBS: .WORD   0               ;Number of blocks in IND.SAV data segment
 65 001566  000000                  USRBAS: .WORD   0               ;Phys 64-byte block # of TSUSR overlay
 66 001570  000000                  MSGBAS: .WORD   0               ;Phys 64-byte block # of TSMSG overlay
 67 001572  000000                  WINBAS: .WORD   0               ;Phys 64-byte block # of TSWIN overlay
 68 001574  000000                  LOKBAS: .WORD   0               ;Phys 64-byte block # of TSLOCK overlay
 69 001576  000000                  CSHBAS: .WORD   0               ;Phys address of TSCASH code
 70 001600  000000                  TIOBAS: .WORD   0               ;Phys address of TSTIOX code
 71 001602  000000                  LOKMEM: .WORD   0               ;Phys 64-byte block # of rec locking data area
 72 001604  000000                  LOKCSH: .WORD   0               ;Phys 64-byte block # of shared file cache buf
 73 001606  000000                  NUMDCD: .WORD   NUMDC           ;Num of shared file data cache entries
 74 001610  000036                  NUMCDB: .WORD   MAXSFC          ;Number of free shared file channels
 75 001612  000000                  SNMSHD: .WORD   0               ;Head of free list of system message buffers
 76 001614  000002                  NMUMB:  .WORD   <NMSNMB-NMSYMB> ;# message buffers available for user access
 77 001616  000000                  CSHHD:  .WORD   0               ;Head of directory cache list
 78 001620  000000                  MONFQH: .WORD   0               ;Head of free list of monitor control blocks
 79 001622  000000                  MIOBHD: .WORD   0               ;Head of mapped I/O control block list
 80 001624  000000                  MIOWHD: .WORD   0               ;Head of mapped I/O wait block list
 81 001626  000000                  MIOSYQ: .WORD   0               ;Pointer to 1st active mapped I/O wait block
 82 001630  000000                  SMONHD: .WORD   0               ;Head of job monitoring requests for all jobs
 83 001632  000000                  SFCB:   .WORD   0               ;Start of spool file control block area
 84 001634  000000                  SFCBND: .WORD   0               ;End of spool file control block area
 85 001636  000000                  SFCBFH: .WORD   0               ;Head of free spool file control block list
 86 001640  000024                  NSPLFL: .WORD   SPLNF           ;Number of spool files
 87 001642  000764                  NSPLBL: .WORD   SNDBX           ;Number of blocks in spool file
 88 001644  000000                  NFRESB: .WORD   0               ;Number of public spool file blocks
 89 001646  000000                  SHRRCB: .WORD   0               ;Pointer to base of global RCB area
 90 001650  000000                  SHRRCN: .WORD   0               ;Pointer to end of global RCB area
 91 001652  000000                  INSTBL: .WORD   0               ;Pointer to base of INSTALL table
 92 001654  000000                  INSTBN: .WORD   0               ;Pointer past end of INSTALL table
 93 001656  000000                  ABRTOV: .WORD   0               ;Rad50 name of overlay during trap
 94 001660  000012                  VMXCSH: .WORD   MAXCSH          ;Max number of cached devices
 95 001662  000000                  CSHDEV: .WORD   0               ;Start of area with device cache blocks
 96 001664  000000                  CSHDVN: .WORD   0               ;End of area with device cache blocks
 97 001666  000000                  SCPFHD: .WORD   0               ;Head of free list of swap command packets
 98 001670  177777                  LDDEVX: .WORD   -1              ;Device index number of "LD" device
 99 001672  177777                  CLDEVX: .WORD   -1              ;Device index number of "CL" device
100 001674  177777                  C1DEVX: .WORD   -1              ;Device index number of "C1" device
101 001676  000000  000000  000000  BOTDEV: .WORD   0,0,0,0         ;Device spec for device being booted from
     001704  000000
102 001706  000000                  BOTUNI: .WORD   0               ;Unit # of device being booted from
103 001710  000000                  BOTCSR: .WORD   0               ;CSR of device being booted from
104 001712  000000                  SPOLID: .WORD   0               ;Last spool file ID number
105 001714  000000                  UMSYTP: .WORD   0               ;Address of top of unmapped system space
106 001716  000001                  IOABFL: .WORD   IOABT           ;1==>Do I/O abort, 0==>Do I/O wait
107 001720  000000G                 DEFBAS: .WORD   TSDEFS
108 001722  114716                  SYNAME: .RAD50  /XXN/           ;Actual name of SY physical device
109 001724  075250  100020  101704  UCLNAM: .RAD50  /SY TSXUCLSAV/  ;Name of TSXUCL program
     001732  073376
110 001734  000000                  UCLBLK: .WORD   0               ;# blocks in TSXUCL data file for each job
111 001736  075250  102405  057760  UKMNAM: .RAD50  /SY UKMON SAV/  ;Name of user-provided TSKMON command processr
     001744  073376
```

```
                                        PIDPTR: .WORD   0               ;Pointer to clock-driven PI handler routine
                                        PROSLT: .BLKW   9.              ;ID # of device in each PRO option slot
                                        VIDCSR: .WORD   0               ;Address of PRO video CSR
                                        VDMTCR: .WORD   DMPTCR          ;Transmitter control reg addr for dump device
                                        MODDAT: .WORD   0               ;Date last modified by TSXMOD
                                        MODTIM: .WORD   0               ;Time (3-sec) last modified by TSXMOD
                                        HANRCB: .WORD   0               ;Pointer to start of handler RCB area
                                        ;
                                        ;  Data for generalized data cache
                                        ;
                                        CA$BLK: .WORD   0               ;Block number associated with cache entry
                                        CA$DVU: .WORD   0               ;Device and unit # associated with entry
                                        CA$WCT: .WORD   0               ;Number of words in entry
                                        CA$UFL: .WORD   0               ;LRU chain forward link
                                        CA$UBL: .WORD   0               ;LRU chain backward link
                                        CA$HFL: .WORD   0               ;Hash chain forward link
                                        CA$HBL: .WORD   0               ;Hash chain backward link
                                        CA$HSH: .WORD   0               ;Hash chains list head vector
                                        CSHBFP: .WORD   0               ;64-byte block number of buffer area
                                        CSHLRU: .WORD   0               ;Pointer to least-recently-used entry
                                        CSHMRU: .WORD   0               ;Pointer to most-recently-used entry
                                        CSHFHD: .WORD   0               ;Head of cache block free list
                                        CCBHD:  .WORD   0               ;Head of cache control block free list
                                        CASTRO: .WORD   0,0             ;Total # reads from mounted devices
                                        CASTBR: .WORD   0,0             ;Total # blocks read from mounted devices
                                        CASCBR: .WORD   0,0             ;Number of blocks that were read from cache
                                        CASTWO: .WORD   0,0             ;Total # writes to mounted devices
                                        CASTBW: .WORD   0,0             ;Total # blocks written to mounted devices
                                        CASCUP: .WORD   0,0             ;Number of blocks moved into data cache
                                        ;  Entry point vector for caching module
                                        CSHVEC:
                                        CSHINI: .WORD   0               ;-
                                        CSHIO:  .WORD   0               ;-
                                        CSHCLN: .WORD   0               ;-
                                        CSHFIN: .WORD   0               ;-
                                                .WORD   -1              ;- End of pointer vector
                                        ;
                                        ;  Entry point vector for record locking module
                                        ;
                                        LOKVEC:
                                        LOKINI: .WORD   0               ;-
                                        DOOPAP: .WORD   0               ;-
                                        DOCOPN: .WORD   0               ;-
                                        SFSVST: .WORD   0               ;-
                                        SFRSST: .WORD   0               ;-
                                        DORLK:  .WORD   0               ;-
                                        DOTLK:  .WORD   0               ;-
                                        DOCULK: .WORD   0               ;-
                                        DOULK1: .WORD   0               ;-
                                        DOSFCK: .WORD   0               ;-
                                        SFCLS:  .WORD   0               ;-
                                        SFWRIT: .WORD   0               ;-
                                        CLSCDB: .WORD   0               ;-
                                        DCRD1:  .WORD   0               ;-
                                        DCRD2:  .WORD   0               ;-
                                                .WORD   -1              ;- End of pointer vector
                                        ;
```

```
                                        ;  Misc byte data
                                        ;
002140  000             VSYDMP: .BYTE   SYSDMP          ;Generate dump on crash if non-zero
002141  000             VDMKTP: .BYTE   DMPKTP          ;Crash on any kernel trap if non-zero
002142  001             VSWPFL: .BYTE   SWAPFL
002143  001             VBUSTP: .BYTE   BUSTYP
002144  000             VINABT: .BYTE   INIABT
002145  001             VUXIFL: .BYTE   UXIFLG
002146  001             VU$CL:  .BYTE   U$CL
002147  002             VUCLOR: .BYTE   UCLORD
002150  001             VLDSYS: .BYTE   LDSYS
002151  001             VSLEDT: .BYTE   SLEDIT
002152  000             VDBFLG: .BYTE   DBGFLG
002153  023             VPRILO: .BYTE   PRILOW
002154  120             VPRIHI: .BYTE   PRIHI
002155  062             VPRIDF: .BYTE   PRIDEF
002156  012             VPRIVR: .BYTE   PRIVIR
002157  035             VTSLCH: .BYTE   TSLICH
002160  027             VVLSCH: .BYTE   VLSWCH
002161  002             VVPWCH: .BYTE   PWCH
002162  034             VCXTRM: .BYTE   CCXTRM
002163  001             VCXCTL: .BYTE   CCXCTL
002164  003             VEDIT:  .BYTE   EDITOR
002165  001             VMIOBF: .BYTE   MIONBF
002166  010             VMIOSZ: .BYTE   MIOBSZ
002167  000             VUSPHN: .BYTE   PHONE           ;0=local if no DCD;1=always mon DCD if $phone
002170  000             MAPUSR: .BYTE   0               ;Number of job memory mapping is set up for
002171  000             LINNUM: .BYTE   0
002172  000             MUXNUM: .BYTE   0
002173  000             NUMON:  .BYTE   0
002174  000             PVON:   .BYTE   0
002175  000             TOTON:  .BYTE   0
002176  000             PROFLG: .BYTE   0               ;Non-zero ==> Running on PRO-350
002177  000             STPFLG: .BYTE   0
002200  000             UBUSMP: .BYTE   0               ;1==>Do Unibus mapping
002201  000             SR3FLG: .BYTE   0               ;NON-ZERO==>MEMORY MANAGEMENT REG 3 PRESENT
002202  000             MEM256: .BYTE   0               ;Non-zero==>machine has at least 256kb
002203  000             MIOFLG: .BYTE   0               ;Non-zero==>I/O mapping needed for some device
002204  000             NSPLDV: .BYTE   0               ;Number of installed spooled devices
002205  000             CLVERS: .BYTE   CLVRSN          ;CL handler version number
002206  000             LDVERS: .BYTE   0               ;LD translation table format (1<RTV5.4=<2)
002207  041     200     SYPSPR: .ASCII  /!/<200>        ;Prompt for system password
                                .EVEN
                        ;  System time counters
002212  000000          TMTOTH: .WORD   0               ;Total uptime (0.1 second units)
002214  000000          TMTOTL: .WORD   0
002216  000000          TMUSRH: .WORD   0               ;Time spent in user jobs
002220  000000          TMUSRL: .WORD   0
002222  000000          TMSWTH: .WORD   0               ;Swap-wait time
002224  000000          TMSWTL: .WORD   0
002226  000000          TMIOH:  .WORD   0               ;Time user i/o is active
002230  000000          TMIOL:  .WORD   0
002232  000000          TMSWPH: .WORD   0               ;Time swapping is active
002234  000000          TMSWPL: .WORD   0
002236  000000          TMIOWH: .WORD   0               ;Time system is doing i/o-wait
002240  000000          TMIOWL: .WORD   0
002242  000000          TMIDLH: .WORD   0               ;Idle time
```

```
    226 002244  000000                TMIDLL: .WORD   0
    227                               ;
    228                               ;  Shared file data cache statistics counters
    229                               ;
    230 002246  000000                DCTOTU: .WORD   0               ;Total number of cache hits since last divisn
    231 002250  000000                DCTRD:  .WORD   0               ;Total number of reads from shared files
    232 002252  000000                DCCRD:  .WORD   0               ;Number of reads satisfied by data in cache
    233 002254  000000                DCTWR:  .WORD   0               ;Total number of writes to shared files
    234 002256  000000                DCCWR:  .WORD   0               ;Number of writes that update cache
    235                               ;
    236 002260  000000                NUMDEV: .WORD   0               ;Byte index to last entry in device tables
    237 002262  000000                FREIOQ: .WORD   0               ;Head of i/o queue element chain
    238                               ;
    239                               ;  Define mux tables for DZ11's and DH11's.
    240                               ;
    241                                       .MACRO  MXTBL   NAME
    242                                       .NLIST
    243                               NAME    =       .-2
    244                                       .GLOBL  NAME
    245                                       .REPT   MAXMUX
    246                                       .WORD   0
    247                                       .ENDR
    248                                       .LIST
    249                                       .ENDM   MXTBL
    250                               ;
    251 002264                                MXTBL   MXTYPE          ;DZ11 & DH11 type of mux (CDX$DZ or CDX$DH)
    252 002304                                MXTBL   MXCSR           ;DZ11 Control Status Register
    253 002324                                MXTBL   MXLPR           ;DZ11 Line Parameter Register
    254 002344                                MXTBL   MXTCR           ;DZ11 Transmit Control Register
    255 002364                                MXTBL   MXDTR           ;DZ11 Data Terminal Ready
    256 002404                                MXTBL   MXTBUF          ;DZ11 Transmitter Buffer Register
    257 002424                                MXTBL   MXSBRK          ;DZ11 Shadow register for hardware BRK reg.
    258 002444                                MXTBL   MXCAR           ;DZ11 Carrier Detect
    259 002464                                MXTBL   MXVEC           ;DZ11 & DH11 Vector address
    260 002504                                MXTBL   MXLNT           ;DZ11 & DH11 Addr of table to map mux # to Lin
    261 002524                                MXTBL   MH$BRK          ;DH11 Break control register
    262 002544                                MXTBL   MH$LPR          ;DH11 Line Parameter Register
    263 002564                                MXTBL   MH$PBR          ;DH11 Previous value of BAR register
    264 002604                                MXTBL   DM$CSR          ;DH11(DM11) Control Status Register
    265 002624                                MXTBL   DM$LSR          ;DH11(DM11) Line Status Register
    266 002644                                MXTBL   DM$VEC          ;DH11(DM11) Address of DM11 interrupt vector
    267         002322'               MXRBUF  =       MXLPR           ;DZ11 Receiver Buffer Register
    268         002402'               MXRING  =       MXTBUF          ;DZ11 Ring indicator flags
    269         002442'               MXBRK   =       MXCAR           ;DZ11 Break control flags
    270                               ;  Equates for DH11 control registers
    271         002302'               MH$SCR  =       MXCSR           ;DH11 System Control Register
    272         002322'               MH$RCR  =       MXRBUF          ;DH11 Received Character Register
    273         002342'               MH$CAR  =       MXTCR           ;DH11 Current Address Register
    274         002362'               MH$BCR  =       MXDTR           ;DH11 Byte Count Register
    275         002402'               MH$BAR  =       MXTBUF          ;DH11 Buffer Active Register
    276         002442'               MH$SSR  =       MXCAR           ;DH11 Silo Status Register
    277                               ;  Equates for DHV11 control registers
    278         002302'               VH$CSR  =       MH$SCR          ;DHV11 Control and Status Register
    279         002322'               VH$DBR  =       MH$RCR          ;DHV11 Data Buffer Register
    280         002542'               VH$LPR  =       MH$LPR          ;DHV11 Line Parameter Register
    281         002622'               VH$LSR  =       DM$LSR          ;DHV11 Line Status Register
    282         002602'               VH$LCR  =       DM$CSR          ;DHV11 Line Control Register
```

```
                002342'                 VH$BA1  =       MH$CAR          ;DHV11 Buffer Address register 1
                002442'                 VH$BA2  =       MH$SSR          ;DHV11 Buffer Address register 2
                002362'                 VH$BCR  =       MH$BCR          ;DHV11 Byte Count Register
                                        ;
                                        ;  Generate FORK request blocks.
                                        ;
002664  002670'                 FREFRK: .WORD   FRKLST           ;Head of free list
002666  000000                  FRKINI: .WORD   0                ;Pointer to fork blocks in init area
002670                          FRKLST:
                000004                  .REPT   FRKGEN           ;Gen in a few static fork blocks
                                        .WORD   .+22.            ;Link to next block in free list
                                        .WORD   0,0,0,0,0,0,0,0,0,0
                                        .ENDR
003020  000000  000000  000000          .WORD   0,0,0,0,0,0,0,0,0,0,0 ;Last block with 0 forward link
003026  000000  000000  000000
003034  000000  000000  000000
003042  000000  000000
                                        ;
                                        ;  Symbolic equates for QBUS and UNIBUS machines.
                                        ;
                000001                  QBUS    =       1
                000000                  UNIBUS  =       0
                                        ;
                                        ;  Generate the memory size limit checking certain restrictions.
                                        ;  On non-extended machines, allow 256.Kb - 8.Kb I/O page
                                        ;  On extended machines, allow 4096.Kb - 256.Kb I/O page
                                        ;
                                                .MACRO  MEMORY  SIZE
                                                SIZMEM  = SIZE
                                                .IF     LE, SIZMEM
                                                        SIZMEM = 3840.
                                                .ENDC
                                                .IF     GT, SIZMEM-3840.
                                                        SIZMEM = 3840.
                                                .ENDC
                                                .IF     LT, <SIZMEM - 96.>
                                        .ERROR  ;Memory size limit too small for running TSX-Plus
                                                .ENDC
                                        ;  Allocate the memory size to examine.
                                                .WORD   SIZMEM*20
                                                .ENDM   MEMORY
                                        ;
                                        ;  Memory management tables
                                        ;
003046                          MAPSIZ: MEMORY  MEMSIZ           ;PAR value of physical memory cutoff
003050  000000                  BASMAP: .WORD   0                ;Pointer to base of memory map table
003052  000000                  LOMAP:  .WORD   0                ;Pointer to 1st user page in MEMMAP
003054  000000                  HIMAP:  .WORD   0                ;Pointer above top user page in memmap
003056  000000                  MAPPAR: .WORD   0                ;Value to map PAR 5 to mem allocation table
003060  000000                  FREPGS: .WORD   0                ;# free pages
003062  000000                  JCXPGS: .WORD   0                ;# pages needed for job context block
003064  000000                  MXJMEM: .WORD   0                ;Max # K-bytes a job may use
003066  000000                  DFJMEM: .WORD   0                ;Default # K-bytes a job may use
003070  000000                  MXJADR: .WORD   0                ;Address above top of largest job space
003072  000000                  SMRSIZ: .WORD   0                ;# 64-byte blocks allocated to system overlays
003074  000000                  MHNSIZ: .WORD   0                ;# 64-byte blocks allocated for mapped handler
003076  000000                  SRTSIZ: .WORD   0                ;# 64-byte blocks allocated for shared run-tim
```

```
                                CSHSIZ: .WORD   0               ;# 64-byte blocks allocated for data cache
                                ;
                                ;  Start a CSECT to hold shared run-time descriptor blocks
                                ;
                                        .CSECT  RDBSEC          ;CSECT for RDB entries
                                RDBSEC:
                                RDB:                            ;Define base of RDB entries
                                        .CSECT  TSGEN           ;Go back to standard TSGEN CSECT
                                ;
                                ;  Symbolic equates for system editor names.
                                ;  Note these equates must match those in TSDEFS.
                                ;
                                EDIT    =       1
                                TECO    =       2
                                KED     =       3
                                K52     =       4
                                ;
                                ;  Symbolic equates for UCL order
                                ;
                                FIRST   =       1
                                MIDDLE  =       2
                                LAST    =       3
                                ;
                                ;  Symbolic names used to define line transmit/receive speeds.
                                ;
                                S50     =       0               ;50     baud
                                S75     =       1               ;75     baud
                                S110    =       2               ;110    baud
                                S134.5  =       3               ;134.5  baud
                                S150    =       4               ;150    baud
                                S300    =       5               ;300    baud
                                S600    =       6               ;600    baud
                                S1200   =       7               ;1200   baud
                                S1800   =       10              ;1800   baud
                                S2000   =       11              ;2000   baud
                                S2400   =       12              ;2400   baud
                                S3600   =       13              ;3600   baud
                                S4800   =       14              ;4800   baud
                                S7200   =       15              ;7200   baud
                                S9600   =       16              ;9600   baud
                                S19200  =       17              ;19200  baud
                                ;
                                ;  Symbolic names for parity codes
                                ;
                                EVEN    =       040000          ;Even parity
                                ODD     =       140000          ;Odd parity
                                NONE    =       000000          ;No parity
```

```
                                ;-----------------------------------------------------------
                                ;  The following macro define the device handler tables.
                                ;  There are two psects use in the device definition - one
                                ;  allocates and defines the rad50 device name - the second
                                ;  defines the handler attributes.
                                ;
                                        .MACRO  DEVBEG          ;DEFINE THE DEVICE GLOBAL ENTRIES
                                        .CSECT  DNAME           ;DEFINE THE DEVICE NAME PSECT
                                AUTHAN:                         ;GLOBAL LABEL FOR DEVICE NAMES
                                        .CSECT  DTYPE           ;DEFINE THE DEVICE TYPE PSECT
                                DTYPE:
                                        .CSECT  TSGEN
                                        .ENDM   DEVBEG

                                ;-------------------------------------------------------------------------------
                                ;  The following flag definitions must match the TSDEFS definitions.
                                ;

000001                          DMA     =       1       ;DX$DMA - This is a DMA device
000002                          MAPIO   =       2       ;DX$MAP - 18-bit controller -- may require mapped I/O
000004                          EVNBUF  =       4       ;DX$EBA - Buffer must be on even byte boundary
000010                          NOCACHE =       10      ;DX$NCA - Do not do caching for this device
000020                          NOMOUNT =       20      ;DX$NMT - Do not allow mounts for this device
000040                          REQALC  =       40      ;DX$RAL - Require device to be allocated before use
000100                          MAPH    =       100     ;DX$MPH - Map the handler for this device
000200                          NOMAPH  =       200     ;DX$NHM - Do not map the handler for this device
000400                          HANBUF  =       400     ;DX$IBH - Handler contains internal I/O buffer
001000                          HNSPDO  =       1000    ;DX$NRD - Do .SPFUN to tell handler about dir ops
002000                          NOSET   =       2000    ;DX$NST - Do not reload handler after SET

000000                          NODMA   =       0               ;This is not a DMA device
000000                          NONDMA  =       NODMA

                                ;-------------------------------------------------------------------------------
                                ;  The DEVDEF macro defines the device name and allocates
                                ;  table entries for the device name and the device attributes.
                                ;
000001                          DVNUM = 1

                                        .MACRO  DEVDEF  DEVNAM,DFLG1,DFLG2,DFLG3,DFLG4,DFLG5,DFLG6,DFLG7,DFLG8,DFLG9

                                DVNUM = DVNUM + 1               ;Increment the device number
                                DVFLG   =       0               ;Get device flags in DVFLG

                                        .IF     LT, <MAXDEV-2 - DVNUM>  ;Check the maximum devices allowed
                                .ERROR  1;More devices defined than MAXDEV
                                        .MEXIT
                                        .ENDC

                                ;  Accumulate flags for the device definition

                                        .IF     NB DFLG1        ;Check if argument exists
                                DVFLG = DVFLG!DFLG1             ;Include in device attributes
                                        .ENDC ;NB DFLG1

                                        .IF     NB DFLG2        ;Check if argument exists
                                DVFLG = DVFLG!DFLG2             ;Include in device attributes
```

```
                        .ENDC ;NB DFLG2

                        .IF     NB DFLG3        ;Check if argument exists
DVFLG = DVFLG!DFLG3             ;Include in device attributes
                        .ENDC ;NB DFLG3

                        .IF     NB DFLG4        ;Check if argument exists
DVFLG = DVFLG!DFLG4             ;Include in device attributes
                        .ENDC ;NB DFLG4

                        .IF     NB DFLG5        ;Check if argument exists
DVFLG = DVFLG!DFLG5             ;Include in device attributes
                        .ENDC ;NB DFLG5

                        .IF     NB DFLG6        ;Check if argument exists
DVFLG = DVFLG!DFLG6             ;Include in device attributes
                        .ENDC ;NB DFLG6

                        .IF     NB DFLG7        ;Check if argument exists
DVFLG = DVFLG!DFLG7             ;Include in device attributes
                        .ENDC ;NB DFLG7

                        .IF     NB DFLG8        ;Check if argument exists
DVFLG = DVFLG!DFLG8             ;Include in device attributes
                        .ENDC ;NB DFLG8

                        .IF     NB DFLG9        ;Check if argument exists
DVFLG = DVFLG!DFLG9             ;Include in device attributes
                        .ENDC ;NB DFLG9

;  Enter the device name into the table defining handlers to load on startup

                        .CSECT  DNAME
                        X = .
                        .RAD50  /'DEVNAM'/      ;Include the device name in the PSECT
                        .IF     NE, <.-X-2>
.ERROR  2;Incorrect device name specified
                        .MEXIT
                        .ENDC

;  Enter the device specification flag into handler flags table

                        .CSECT  DTYPE
                        .WORD   DVFLG           ;Include the device type in the PSECT
                        .CSECT  TSGEN

                        .ENDM   DEVDEF

;-------------------------------------------------------------
;  The DEVEND macro allocates the remainder of the table entries.
;
                        .MACRO  DEVEND
L                       = <MAXDEV-2-DVNUM>
                        .IF     GT,L
                        .REPT   L
                        DEVDEF  <$$ >
                        .ENDR
```

```
115                                             .ENDC    ;GT,L
116                                             .CSECT   DNAME
117                                     AHEND:
118                                             .CSECT   TSGEN
119                                             .ENDM    DEVEND
```

```
                ;-----------------------------------------------------------
                ;  The OB macro creates a table with NLINES entries
                ;  and defines the name of the table to be
                ;  1 word in front of the start of the table.
                ;  The name if globally defined.
                ;
                        .MACRO  OB      NAME
                        .NLIST
                NAME    =       .-2
                        .GLOBL  NAME
                        .REPT   NLINES
                        .WORD   0
                        .ENDR
                        .LIST
                        .ENDM   OB

                ;-----------------------------------------------------------
                ;  The OBP macro is similar to OB except it
                ;  generates only NPL (# of primary lines) entries
                ;  instead of NLINES entries.
                ;
                        .MACRO  OBP     NAME
                        .NLIST
                NAME    =       .-2
                        .GLOBL  NAME
                        .REPT   NPL
                        .WORD   0
                        .ENDR
                        .LIST
                        .ENDM   OBP

                ;-----------------------------------------------------------
                ;  The OBH macro is similar to OB except it
                ;  generates  TNHL (# lines requiring hardware control tables) entries
                ;  instead of NLINES entries.
                ;
                        .MACRO  OBH     NAME
                        .NLIST
                NAME    =       .-2
                        .GLOBL  NAME
                        .REPT   TNHL
                        .WORD   0
                        .ENDR
                        .LIST
                        .ENDM   OBH

                ;-----------------------------------------------------------
                ;  The OBT macro is similar to OB except it
                ;  generates  NPL+NSL+NDL+NIOL entries
                ;  instead of NLINES entries.
                ;
                        .MACRO  OBT     NAME
                        .NLIST
                NAME    =       .-2
                        .GLOBL  NAME
                        .REPT   NPL+NSL+NDL+NIOL
                        .WORD   0
```

```
                        .ENDR
                        .LIST
                        .ENDM   OBT

;-----------------------------------------------------------------------
;  The TBLDEF macro is called once to define table
;  space needed by all of the lines.
;  It has four arguments:
;  Argument 1 is the number of primary (real) lines.
;  Argument 2 is the number of subprocesses.
;  Argument 3 is the number of detached lines.
;  Argument 4 is the number of dedicated CL lines.
;
        .MACRO  TBLDEF  ANPL,ANSL,ANDL,ANIOL
        .NLIST  MD
NPL     =       ANPL            ;# of primary lines
NSL     =       ANSL            ;Number of subprocesses
NDL     =       ANDL            ;Number of detached lines
NIOL    =       ANIOL           ;Number of dedicated CL lines
NLINES  =       NPL+NSL+NDL     ;Total number of jobs
;
;  Make sure the total number of CL units does not exceed 16
;
        .IF     GT <NIOL-16.>
.ERROR ;You cannot have more than 16 CL units
NIOL    =       16.             ;Reduce number to 16.
        .ENDC
        .IF     GT <<NIOL+CLXTRA>-16.>
.ERROR ;You cannot have more than 16 CL units
CLXTRA  =       16.-NIOL                ;Reduce extra units if total > 16
        .ENDC
CLTOTL  =       NIOL+CLXTRA
;
;  Set up number of lines variables.
;  The lines are numbered in the following order:
;    Primary lines.
;    Detached job lines.
;    Subprocesses.
;    Dedicated CL lines.
;
LSTPL   =       2*NPL           ;Last primary line index
FSTDL   =       LSTPL+2         ;First detached line
LSTDL   =       LSTPL+<2*NDL>   ;Last detached line
FSTSL   =       LSTDL+2         ;Index to first subprocess
NLIN2   =       2*NLINES        ;Index to last time-sharing line
LSTSL   =       NLIN2           ;Index to last subprocess
FSTIOL  =       LSTSL+2         ;Index to first CL line
LSTIOL  =       FSTIOL+<2*<NIOL-1>> ;Index to last CL line
LSTLIN  =       2*<NPL+NSL+NDL+NIOL> ;Index number of last line
        .IF     EQ,NIOL         ;If there are no CL lines
TNHL    =       NPL             ;Total number of lines with hardware-ctrl tbls
        .IFF                    ;If there are CL lines
TNHL    =       NPL+NSL+NDL+NIOL;Total number of lines with hardware-ctrl tbls
        .ENDC
LSTHL   =       TNHL*2          ;Index # of last hardware line
;
;  Define number of slots in job swap file if SWPSLT=0
```

```
                        ;
                                .IF     EQ,SWPSLT
SWPSLT  =               NLINES          ;Default to one slot for each job
                                .ENDC   ;EQ,SWPSLT
                                .IF     GT,<SWPSLT-NLINES> ;Never need more slots than lines
SWPSLT  =               NLINES
                                .ENDC   ;GT,<SWPSLT-NLINES>
                                .IF     EQ,SWAPFL       ;If this is a non-swapping system
SWPSLT  =               0               ;No swap slots needed
                                .ENDC   ;EQ,SWAPFL
                        ;
                        ;  Define line tables.
                        ;
                                OB      LQLINK          ;Link for execution queues
                                OB      LSTATE          ;Current execution state
                                OB      LBSPRI          ;Job base priority value (byte)
LPRI    =               LBSPRI+1        ;Current job priority (byte)
                                OB      LPARNT          ;Index number of parent job
                                OBT     LSW             ;Line status word
                                OB      LSW2            ;Additional line status
                                OB      LSW2S           ;Copy of LSW2 used for reset on prog exit
                                OBH     ILSW2           ;Initial values for LSW2
                                OBT     LSW3            ;Additional line status flags
                                OB      LSW4            ;Additional line status flags
                                OBT     LSW5            ;More line status flags
                                OBT     LSW6            ;Line status table # 6
                                OB      LSW7            ;Line status table # 7
                                OB      LSW8            ;Line status table # 8
                                OB      LSW9            ;Line status table # 9
                                OBT     LSW10           ;Line status table # 10
                                OB      LSW11           ;Line status table # 11
                                OBT     LCLUNT          ;CL unit index number if connected as CL line
                                OBP     ITRMTP          ;Initial terminal type code
                                OBT     LTRMTP          ;Current terminal type code
                                OBH     LNAME           ;Descriptive name for line
                                OB      LMEMIN          ;# pages of memory needed to inswap job
                                OB      LPARBS          ;PAR base address for job
                                OB      LCXPAR          ;Value for KPAR6 to map to job context block
                                OB      LNBLKS          ;# pages of memory currently assigned to job
                                OB      LNSBLK          ;# pages of memory used by PLAS regions
                                OB      LTTPAR          ;Physical memory PAR value for terminal buffer
                                OB      LQUAN           ;Job's execution quantum
                                OB      LITIME          ;Time job is held in "interactive" state
                                OB      LIOHLD          ;Hold time for I/O starts while starting swap
                                OB      LMINQ           ;Minimum core-residency time
                                OB      LHIPCT          ;Controls # high-prio quantum periods job gets
                                OB      LIOCNT          ;# active i/o operations for job
                                OB      LBASE           ;Base page # assigned to job
                                OBH     LHIRBB          ;Start of silo input ring buffer
                                OBH     LHIRBE          ;End of silo input ring buffer
                                OBH     LHIRBA          ;Allocated size of silo input ring buf
                                OBH     LHIRBS          ;Free space in silo input ring buffer
                                OBH     LHIRBP          ;Pointer where to store next char in buffer
                                OBH     LHIRBG          ;Pointer where to get next char from buffer
                                OBH     LHIRBC          ;Autoflow control stop/start char count limits
                                OBT     LINSIZ          ;Size of input character buffer
                                OB      LINBUF          ;Start of input buffer
```

```
                OB      LINEND          ;End of input buffer
                OB      LINNXT          ;Where next input char goes
                OB      LINPNT          ;Where to get next char read
                OB      LINCNT          ;# of chars in input buffer
                OB      LINSPC          ;# free bytes in input buffer
                OB      LACTIV          ;# of activation chars pending
                OB      LAFSIZ          ;Field width for activation condition
                OB      LFWLIM          ;Field width limit
                OB      LSTACT          ;Position of last activation char
                OB      LINCUR          ;Pos of cursor at start of line
                OBT     LOTSIZ          ;Size of output buffer
                OB      LOTBUF          ;Start of output buffer
                OB      LOTEND          ;End of output buffer
                OB      LOTNXT          ;Place to put next output char
                OB      LOTPNT          ;Place to get next output char
                OB      LOTSPC          ;Space left in output buffer
                OB      LWINDO          ;Pointer to current display window block
                OBH     LCDTYP          ;Type of communications device (CDX$xxx)
                OBH     LINIR           ;Terminal input service routine
                OBH     LOUTIR          ;Terminal output service routine
                OBH     INVEC           ;Input interrupt vector loc
                OBH     RSR             ;Receiver status register address
                OBH     RBR             ;Receiver buffer register
                OBH     TSR             ;Transmitter status register
                OBH     TBR             ;Transmitter buffer register
                OBH     LDHB1B          ;Base of DMA buffer 1
                OBH     LDHB1P          ;Pointer into DMA buffer 1
                OBH     LDHB2B          ;Base of DMA buffer 2
                OBH     LDHB2R          ;Remaining byte count for buffer 2
                OBH     LDHB2S          ;Suspended pointer for buffer 2
                OBH     LCXTBL          ;Pointer to character translation table
                OBP     LSECPT          ;Pointer to secondary line # table
                OBP     LXCL            ;CL unit to which line is cross connected
                OB      LCMPL           ;Head of chain of completion requests for job
                OB      LCMQHD          ;Queue head for completed message requests
                OB      LMONHD          ;Queue head for job monitor blocks
                OB      LSUCF           ;Start-up command file
                OB      LSWPBK          ;Block # in swap file
                OB      LJSW            ;User's JSW
                OB      LEMTPC          ;PC of last user-mode emt
                OB      LSCCA           ;.SCCA control word address
                OB      LSPND           ;.SPND counter for job
                OB      LBRKCQ          ;Break character completion queue entry
                OB      LTTCR           ;Completion routine for TT input activation
                OB      LBRKCH          ;Break character for line
                OB      LCOL            ;Current column position
                OBP     LMSGBF          ;Send message pointer
                OB      LSNDCH          ;Last char sent
                OB      LESRTN          ;Echo suppression routine
                OB      LESCHR          ;Echo suppression char code
                OB      LRBFIL          ;Rubout filler for line
                OB      LTSCMD          ;Pending special action command
                OB      LNSPAC          ;# of special activation chars
                OB      LSPACT          ;Point to special actv char tbl
                OB      LPROJ           ;Project #
                OB      LPROG           ;Programmer #
                OB      LCPUHI          ;High-order CPU time
```

```
                                        OB      LCPULO          ;Low-order CPU time
                                        OB      LCONTM          ;Connect time
                                        OBP     LCDTIM          ;Lost-carrier disconnect time
                                        OBP     LOFFTM          ;Allowed logoff time before DTR drop for line
                                        OBP     LABTIM          ;Autobaud control timer
                                        OB      LRDTIM          ;TT read timeout
                                        OB      LRTCHR          ;TT read timeout activation character
                                        OB      LSLEPL          ;.TWAIT sleep time for job (low-order)
                                        OB      LSLEPH          ;.TWAIT sleep time for job (high-order)
                                        OBH     LMXNUM          ;Index # of mux controlling line
                                        OBH     LMXPRM          ;Line parameters (speed, parity, stop bits)
                                        OB      LPRG1           ;1st 3 chars of running program name (rad50)
                                        OB      LPRG2           ;2nd 3 chars of running program name (rad50)
                                LUNAME  =       .-12.           ;Offset user name table by size of 1 entry
                                        .BLKB   NLINES*12.      ;Store 12 char user name here
                                LMXLN   =       RBR             ;# of this line within mux group
```

```
                        ;
                        ;  Define subprocess mapping tables
                        ;
LNMAP   =       .-2
        .NLIST  BIN
I       =       0
        .REPT   NPL
I       =       I+2
        .WORD   I
        .ENDR
;  Define LNPRIM table
LNPRIM  =       .-2
I       =       0
        .REPT   NPL+NDL
I       =       I+2
        .WORD   I
        .ENDR
        .IF     NE,NSL
        .REPT   NSL
I       =       I+2             ;Keep count for NIOL if any
        .WORD   0
        .ENDR
        .ENDC
        .IF     NE,NIOL
        .REPT   NIOL
I       =       I+2
        .WORD   I
        .ENDR
        .ENDC
        .LIST   BIN
;
;  Generate interrupt receivers
;
;  Input interrupt vector
        .NLIST  BIN
        .REPT   TNHL
        INCB    LINNUM          ;COUNT UP WHICH LINE INTERRUPTED
        .ENDR
INRECV: JMP     ININT           ;ENTER INTERRUPT SERVICE ROUTINE
; Output interrupt vector
LXX     =       2
OTRECV:
        .REPT   TNHL
        JSR     R4,@#DLINT
        .WORD   LXX
LXX     =       LXX+2
        .ENDR
        .LIST   BIN
        .LIST   MD
        .ENDM   TBLDEF
```

```
                                ;-----------------------------------------------------------------------
                                ;  The CLDEF macro begins a line definition block for a serial communications
                                ;  line that will be used as a dedicated CL line.
                                ;  The CLDEF macro is similar to a LINDEF and can occur inside or outside
                                ;  of a MUXDEF block.
                                ;  The form of the CLDEF macro outside a MUXDEF block is:
                                ;       CLDEF   line_number,vector_address,RSR_address
                                ;  The form of the CLDEF macro inside a MUXDEF block is:
                                ;       CLDEF   line_number,mux_line_number
                                ;
                                        .MACRO  CLDEF   AIOLN,ARG1,ARG2
                                ;
                                ;  Check to make sure the CL unit number is valid
                                ;
                                IOLN    =       AIOLN
                                        .IF     GE <IOLN-CLTOTL>
                                .ERROR ;0 CL unit number exceeds # declared CL units
                                IOLN    =       0
                                        .ENDC
                                ;
                                ;  See if this CL unit has already been assigned to another line
                                ;
                                        .IF     NDF     CLUD'AIOLN
                                CLUD'AIOLN      =       1
                                        .ENDC
                                        .IF     GT <CLUD'AIOLN-2>
                                .ERROR ;CL unit AIOLN used more than once
                                        .ENDC
                                CLUD'AIOLN      =       CLUD'AIOLN+1
                                ;
                                ;  Set flag saying we are doing an CLDEF definition and then invoke LINDEF.
                                ;  Note, the LINEND macro will reset IOLFLG.
                                ;
                                IOLFLG  =       1               ;We are inside CLDEF
                                        LINDEF  ARG1 ARG2
                                        .ENDM   CLDEF
                                ;
                                ;-----------------------------------------------------------------------
                                ;  The LINDEF macro begins a line definition block.
                                ;  A line definition block is required for each primary
                                ;  (real) line.  A line definition block begins with
                                ;  a LINDEF macro call, may include other macro calls
                                ;  such as LFLAGS and must end with a LINEND macro call.
                                ;  there are two arguments to the LINDEF macro:
                                ;  Arg 1 is the input interrupt vector address or mux line #.
                                ;  Arg 2 is the address of the receiver status register.
                                ;  Arg 3 is 'OPERATOR' to specify line is control terminal.
                                ;
000000                          LN      =       0               ;Current line number
000000                          BO      =       0               ;1 if inside LINDEF block
000000                          LX      =       0               ;Line index number
000000                          IOLFLG  =       0               ;1 if inside an CLDEF block
000000                          IOLN    =       0               ;Number of dedicated CL line
000000                          NPLDF   =       0               ;Number of declared primary T/S lines
000000                          NCLDF   =       0               ;Number of CL lines that have been defined
000000                          NDLDF   =       0               ;Number of declared detached jobs
                                ;
```

```
                        .MACRO  LINDEF  AINTAD,ARSR,AOPR
                        .IF     NE BO           ;SEE IF LAST BLOCK LEFT OPEN
                .ERROR  1; Missing LINEND on last line
                        LINEND                  ;CLOSE OFF PREVIOUS BLOCK
                        .ENDC
                BO      =       1               ;SAY WE'RE INSIDE A BLOCK
                NAMDON  = 0                     ;SAY NO NAME DECLARED YET
                CMFDON  = 0                     ;SAY NO SUCF DECLARED YET
                ;
                ;  Update current line #
                ;  and make sure we don't overflow tables
                ;
                        .IF     EQ,IOLFLG       ;If not inside an CLDEF block
                LN      =       LN+1            ;Line counter
                NPLDF   =       NPLDF+1         ;Count number of primary lines
                LX      =       LX+2            ;Line index
                CLX     =       LX
                        .IF     GT <LN-NPL>
                .ERROR  2; More lines than declared with TBLDEF
                        .MEXIT
                        .ENDC   ;GT <LN-NPL>
                        .IFF    ;EQ,IOLFLG      ;If inside an CLDEF block
                CLX     =       FSTIOL+<2*NCLDF>;Get line index # of this line
                NCLDF   =       NCLDF+1         ;Count # of dedicated CL lines
                        .IF     GT <NCLDF-NIOL> ;Don't exceed # CL lines declared in TBLDEF
                .ERROR  0; More CL lines than declared in TBLDEF
                        .MEXIT
                        .ENDC   ;GT <NCLDF-NIOL>
                S       =       .
                .       =       LCLUNT+CLX      ;Store CL unit # into table for this line
                        .WORD   2*IOLN
                .       =       S
                        .ENDC
                ;
                ; *** Do this for DL11 lines only ***
                ;
                        .IF     EQ CURMX        ;True if not within mux definition block
                CURMXL  =       0
                ;
                ;  Set up interrupt vector addresses
                ;
                        .IF     B AINTAD
                        .IF     EQ,IOLFLG
                .ERROR 3; Missing interrupt address (arg 1)
                        .IFF
                .ERROR 3; Missing interrupt address (arg 2)
                        .ENDC
                        .MEXIT
                        .ENDC
                        VECCHK  AINTAD,7
                S       =       .
                .       =       INVEC+CLX
                        .WORD   AINTAD
                .       =       S
                ;
                ;  Set up DL11 register addresses
                ;
```

```
                        .IF     B ARSR
                        .IF     EQ,IOLFLG
                .ERROR  4 ; Missing receiver register address (arg 2)
                        .IFF
                .ERROR 4; Missing receiver register address (arg 3)
                        .ENDC
                        .MEXIT
                        .ENDC
                        SRCHK   ARSR
                S       =       .
                .       =       RSR+CLX
                        .WORD   ARSR
                .       =       RBR+CLX
                        .WORD   ARSR+2
                .       =       TSR+CLX
                        .WORD   ARSR+4
                .       =       TBR+CLX
                        .WORD   ARSR+6
                        .IF     NB AOPR         ;SEE IF THIS IS CONTROL TERMINAL
                .       =       CTRLTT          ;REMBER CONTROL TERMINAL #
                        .WORD   CLX
                        .ENDC
                .       =       S
                ;
                ;  *** Do this for DZ11 and DH11 lines only ***
                ;
                        .IFF
                S       =       .
                .       =       LMXNUM+CLX      ;MUX UNIT NUMBER
                        .WORD   CURMX
                        .IF     B AINTAD
                .ERROR 0;Missing multiplexer line number
                CURMXL  =       0
                        .IFF
                CURMXL  =       AINTAD
                        .ENDC
                .       =       LMXLN+CLX       ;LINE WITHIN MUX
                        .WORD   CURMXL
                        .IF     NB ARSR
                .       =       CTRLTT
                        .WORD   CLX
                        .ENDC
                .       =       S
                        .ENDC
                ;
                ;  *** Do this for all lines ***
                ;
                S       =       .
                .       =       LCDTYP+CLX      ;Communications device type index
                        .WORD   CURCDX
                .       =       S
                ;
                ;  Establish default values in case user doesn't specify
                ;  them inside line definition block.
                ;
                DFLAGS  =       NRMFLG          ;DEFAULT LINE CONTROL FLAGS
                        .IF     EQ,IOLFLG       ;If this is not an CLDEF
```

```
                        DIS     =       DINSPC          ;INPUT BUFFER SIZE
                        DOS     =       DOTSPC          ;OUTPUT BUFFER SIZE
                                .IFF                    ;If this is an CLDEF
                        DIS     =       0               ;Input ring buffer size
                        DOS     =       CLORSZ          ;Output ring buffer size
                                .IIF    LE,DOS  DOS = 32.       ;Don't allow <= 0 size
                                .ENDC
                        ;
                        ;  Establish default values for character silos
                        ;
                        SILSIZ  =       0
                        SILXOF  =       0
                        SILXON  =       0
                        ;
                                .ENDM   LINDEF
```

```
                                ;-----------------------------------------------------------
                                ;  The FLAGS macro is used to set flags in the ILSW2 table.
                                ;  The one argument to flags is the value to be stored
                                ;  in ILSW2.
                                ;
                                        .MACRO  FLAGS   AFLG
                                DFLAGS  =       AFLG            ;SAVE FOR LINEND
                                        .ENDM   FLAGS
                                ;-----------------------------------------------------------------------
                                ;  The TRMTYP macro is used to declare the terminal type.
                                ;
                                        .MACRO  TRMTYP  ATYP
                                        .IF     EQ,IOLFLG       ;Do not do for CL lines
                                S       =       .
                                .       =       ITRMTP+CLX
                                        .WORD   ATYP
                                .       =       S
                                        .ENDC
                                        .ENDM   TRMTYP
                                ;
                                ;-----------------------------------------------------------------------
                                ;  The NAME macro is used within a line definition block to declare
                                ;  a commentary name for the line which is displayed with the
                                ;  SHOW TERMINALS keyboard command.
                                ;
 000000                                 .CSECT  NAMSEC
 000000                         NAMSEC:
 003102                                 .CSECT  TSGEN
                                        .MACRO  NAME    NAMSTR
                                        .CSECT  NAMSEC
                                NAMPTR  =       .
                                        .ASCIZ  \NAMSTR\
                                LNMTOP  = .
                                ;
                                        .CSECT  TSGEN
                                S       =       .
                                .       =       LNAME+CLX
                                        .WORD   NAMPTR
                                .       =       S
                                NAMDON  = 1
                                        .ENDM   NAME
                                ;
                                ;-----------------------------------------------
                                ;  The BUFSIZ macro is used to set the size of
                                ;  the input and output character buffers.
                                ;  Arg 1 = Input buffer size (# of characters)
                                ;  Arg 2 = output buffer size (# of characters)
                                ;
                                        .MACRO  BUFSIZ  AIS,AOS
                                DIS     =       AIS
                                        .IF     NB AOS
                                DOS     =       AOS             ;SET OUTPUT BUFFER SIZE
                                        .ENDC
                                        .ENDM   BUFSIZ
                                ;
                                ;-----------------------------------------------------------------------
                                ;  The SILO macro is used to set up information about the
```

```
                                ;  terminal input character silo.
                                ;   Arg 1 = Size of the silo buffer.
                                ;   Arg 2 = Free space remaining when XOFF is to be sent.
                                ;   Arg 3 = Number of chars remaining when XON is to be sent.
                                ;
                                        .MACRO  SILO    ASIZ,AXOF,AXON
                                SILSIZ  =       ASIZ
                                SILXOF  =       AXOF
                                SILXON  =       AXON
                                        .ENDM   SILO
                                ;
                                ;-------------------------------------------------------
                                ;  The PAGE macro is used to establish the number
                                ;  of lines on a page.
                                ;
                                        .MACRO  PAGE ALINES
                                        .ENDM   PAGE
                                ;
                                ;-------------------------------------------------------
                                ;  The CMDFIL macro is used to declare a command file which
                                ;  is to be executed when the line is started.
                                ;
000000                                  .CSECT  CMFSEC
000000                          CMFSEC:
003102                                  .CSECT  TSGEN
                                        .MACRO  CMDFIL  ARG
                                        .IF     EQ,IOLFLG       ;Only do for non-CL lines
                                        .CSECT  CMFSEC
                                NAMPTR  = .
                                        .ASCIZ  /ARG/
                                CMFTOP  = .
                                        .CSECT  TSGEN
                                S       =       .
                                .       =       LSUCF+CLX
                                        .WORD   NAMPTR
                                .       =       S
                                        .ENDC
                                CMFDON  = 1
                                        .ENDM   CMDFIL
                                ;
                                ;-------------------------------------------------------
                                ;  The LINPRM macro is used to specify parameters
                                ;  for lines.
                                ;  There are three parameters:
                                ;  1. Speed select code.
                                ;  2. Even (0) / Odd (1) parity (No longer used).
                                ;  3. One (0) or two (1) stop bits
                                ;
                                        .MACRO  LINPRM  ASPD,APAR,ASTOP
                                PERR    =       0
                                .IIF    GT,ASPD-17      PERR=1
                                .IIF    GT,APAR-2       PERR=1
                                .IIF    LT,ASTOP-1      PERR=1
                                .IIF    GT,ASTOP-2      PERR=1
                                        .IF     NE,PERR
                                .ERROR  ;Invalid speed, parity or stop-bits parameter in LINPRM
                                        .ENDC
```

```
                                XPAR    =       100
                                CARLEN  =       20
                                        .IF     GT,<APAR-1>
                                XPAR    =       0
                                CARLEN  =       30
                                        .ENDC
                                LSTPRM  =       <ASPD*400>
                                        .ENDM   LINPRM
                                ;
                                ;-----------------------------------------------------------------------
                                ;  The SPEED macro is used to specify baud rates for lines
                                ;  as well as number of data bits and parity selection.
                                ;  The default is 9600 baud with 8. data bits and no parity.
                                ;  The form of the macro is:
                                ;
                                ;       SPEED   speedcode,data_bits,parity
                                ;
                                ;  where   speedcode is selected from the speed code table
                                ;            and is of the form S9600, for example
                                ;    and   data_bits = 7 or 8.
                                ;    and   parity = EVEN, ODD or NONE
                                ;
007000                          LSTPRM  = <S9600*400>!<20000*0>!NONE    ;Default to 4800,8,N
                                ;
                                        .MACRO  SPEED   SPDCOD,NBITS,PARCOD
                                        .IF     DF,S'SPDCOD
                                SPDVAL  =       S'SPDCOD
                                        .IFF    ;DF,S'SPDCOD
                                        .IF     DF,SPDCOD
                                SPDVAL  =       SPDCOD
                                        .IFF    ;DF,SPDCOD
                                .ERROR  0;Invalid speed specified with SPEED macro
                                SPDVAL  =       14.
                                        .ENDC   ;NDF,S'SPDCOD
                                        .ENDC   ;DF,S'SPDCOD
                                        .IF     B,NBITS
                                NDBITS  = 8.
                                        .IFF    ;B,NBITS
                                NDBITS  = NBITS
                                        .IF     LT,<NDBITS-7>
                                .ERROR  ;SPEED macro only accepts 7 or 8. data bits
                                NBDITS  = 8.
                                        .ENDC   ;LT,<NDBITS-7>
                                        .IF     GT,<NDBITS-8.>
                                .ERROR  ;SPEED macro only accepts 7 or 8. data bits
                                NDBITS  = 8.
                                        .ENDC   ;GT,<NDBITS-8.>
                                        .ENDC   ;B,NBITS
                                        .IF     B,PARCOD
                                PARITY  = NONE
                                        .IFF    ;B,PARCOD
                                PARITY  = PARCOD
                                        .IF     NE,PARITY       ;NOT NONE?
                                        .IF     NE,<PARITY-EVEN> ;NOT EVEN?
                                        .IF     NE,<PARITY-ODD>  ;NOR ODD?
                                .ERROR  ;Parity must be EVEN, ODD or NONE in SPEED macro
                                PARITY  = NONE
```

```
                        .ENDC   ;NOT ODD
                        .ENDC   ;NOT EVEN
                        .ENDC   ;EVEN OR ODD
                        .ENDC   ;B,PARCOD
LSTPRM  =               <SPDVAL*400>!<20000*<8.-NDBITS>>!PARITY
                        .ENDM   SPEED
;
;-------------------------------------------------------------
;  The VECCHK macro is called to see if a line vector
;  address is reasoanble.
;
                        .MACRO  VECCHK  VA,MASK
VERR=0
.IIF    LT,VA-60        VERR=1
; Although RT-11 V5.3 reserves 470 and 474, we do not (for now).
.IIF    GE,VA-500       VERR=1
.IIF    NE,VA&MASK      VERR=1
                        .IF     NE,VERR
.ERROR  ;Invalid vector address for this line
                        .ENDC
                        .ENDM   VECCHK
;
;-------------------------------------------------------------------
;  SRCHK macro checks the validity of a receiver status
;  register address
;
                        .MACRO  SRCHK   SR
SRERR=0
.IIF    LT,SR-160000    SRERR=1
.IIF    NE,SR&7         SRERR=1
                        .IF     NE,SRERR
.ERROR  ;Invalid status register address for this line
                        .ENDC
                        .ENDM   SRCHK
```

```
                                ;-----------------------------------------------
                                ;  The LINEND macro is used to close out a line
                                ;  definition block.
                                ;
                                        .CSECT  SASECT
                                SASECT:
                                        .CSECT  TSGEN
                                        .MACRO  LINEND
                                ;  Make sure we're inside a line def block.
                                        .IF     EQ BO
                                .ERROR  6 ; Missing LINDEF for this line
                                BO      =       0
                                        .MEXIT
                                        .ENDC
                                BO      =       0               ;END LINDEF BLOCK
                                ;
                                ;  Make sure NAME and CMDFIL reserve at least 1 byte
                                        .IF     EQ,NAMDON
                                        NAME    <>
                                        .ENDC
                                        .IF     EQ,CMFDON
                                        CMDFIL  <>
                                        .ENDC
                                ;
                                ;  Define input and output character buffer sizes for line
                                ;  Define input buffer
                                S       =       .
                                .       =       LINSIZ+CLX
                                        .WORD   DIS-1           ;DEFINE BUFFER SIZE
                                .       =       S
                                ;  Define output buffer
                                S       =       .
                                .       =       LOTSIZ+CLX
                                        .WORD   DOS             ;DEFINE BUFFER SIZE
                                .       =       S
                                ;
                                ;  Define table for user defined activation characters.
                                ;
                                        .IF     EQ,IOLFLG       ;Only do for non-CL lines
                                S       =       .
                                        .CSECT  SASECT
                                T       =       .
                                        .BLKB   MXSPAC
                                        .CSECT  TSGEN
                                .       =       LSPACT+CLX
                                        .WORD   T
                                .       =       S
                                        .ENDC                   ;End conditional (EQ,IOLFLG)
                                ;
                                ;  Items for primary lines only.
                                ;
                                LF      =       0               ;Assume this is not a primary or CL line
                                        .IF     NE,IOLFLG       ;If doing a CL line definition
                                LF      =       1               ;Treat like primary line
                                        .IFF                    ;If not doing a CL line definition
                                        .IF     LE <LN-NPL>     ;Do only if this is a primary line
                                LF      =       1               ;Gen code
```

```
                        .ENDC                   ;End conditional (LE <LN-NPL>)
                        .ENDC                   ;End conditional (NE,IOLFLG)
                ;
                        .IF     NE,LF           ;Do if primary line or CL line
                ;
                ;   Define line control flags
                ;
                S       =       .
                .       =       ILSW2+CLX
                        .WORD   DFLAGS          ;SET THE FLAGS
                .       =       S
                ;
                ;   Defile silo buffer size information
                ;
                S       =       .
                .       =       LHIRBA+CLX      ;Silo size
                        .WORD   SILSIZ
                .       =       LHIRBC+CLX      ;XOFF/XON control info
                        .BYTE   SILXOF          ;XOFF point
                        .BYTE   SILXON          ;XON point
                .       =       S
                ;
                ;   Define line parameters (Required for DZ11 & DH11 lines, optional for DL11)
                ;
                        .IF     NDF,LSTPRM
                LSTPRM  =       0
                        .IF     NE CURMX
                .ERROR  0;Missing SPEED macro call
                        .ENDC           ;End conditional (NE CURMX)
                        .ENDC           ;End conditional (NDF,LSTPRM)
                S       =       .
                .       =       LMXPRM+CLX
                        .WORD   <LSTPRM>!<CURMXL>
                .       =       S
                ;
                ;   Define subprocess table.
                ;
                        .IF     EQ,IOLFLG       ;Do if not inside CLDEF block
                        .IF     NE MAXSEC
                S       =       .
                .       =       LSECPT+CLX
                        .WORD   S
                .       =       S
                        .REPT   MAXSEC
                        .BYTE   0
                        .ENDR
                        .EVEN
                        .ENDC           ;End conditional (NE MAXSEC)
                        .ENDC           ;End conditional (EQ,IOLFLG)
                ;
                ;   Define chracter translation table
                ;
                        .IF     NE MXTTCT
                S       =       .
                .       =       LCXTBL+CLX
                        .WORD   S
                .       =       S
```

```
                        .REPT   MXTTCT+1
                        .WORD   0
                        .ENDR
                        .EVEN
                        .ENDC           ;End conditional (NE MXTTCT)
;
                        .ENDC           ;End conditional (NE,LF)
;
;  Reset flag that says we are inside an CLDEF block
;
IOLFLG  =       0       ;No longer inside an CLDEF block
                        .ENDM   LINEND
;
;-----------------------------------------------------------------------
;  The CLEND macro is like the LINEND macro except it is used to
;  terminate a communication line definition started with a CLDEF macro.
;
                        .MACRO  CLEND
                        LINEND
                        .ENDM   CLEND
;
;-----------------------------------------------------------------------
;  The DETACH macro is used to define a start-up command file
;  To be run on a detached line when TSX-Plus is started.
;  The one argument to DETACH is the name of the command file.
;
                        .MACRO  DETACH  NAME
                        .IF     NE      BO
.ERROR  1; Missing LINEND on last line
                        LINEND
                        .ENDC
NDLDF   =       NDLDF+1         ;Count number of detached jobs
LN      =       LN+1
LX      =       LX+2
CLX     =       LX
                        .IF     LE,<NDL>
.ERROR  2; DETACH macro declared with no detached lines
                        .MEXIT
                        .ENDC
                        .IF     GT,<LN-NPL-NDL>
.ERROR  2; More lines than declared with TBLDEF
                        .MEXIT
                        .ENDC
;  Store startup command file name
S       =       .
.       =       LSUCF+CLX
                        .WORD   S
.       =       S
                        .IF     NB,<NAME>
                        .ASCIZ  /NAME/
                        .ENDC
                        .REPT   <DETCBS+1-<.-S>>
                        .BYTE   0
                        .ENDR
                        .EVEN
                        .ENDM   DETACH
;
```

```
                                ;-------------------------------------------------------------------------------
                                ;  The SYSPS macro is used to define a system password which may be
                                ;  required to be entered for some lines before the normal logon
                                ;  sequence begins.
                                ;
000000                                  SYSPSS  = 0
                                        .MACRO  SYSPS   STRING
                                        SYSPSS = 1
                                SYPSWD: .ASCIZ  \STRING\
                                        .IF     GT,<21.-<.-SYPSWD>>
                                        .REPT   <21.-<.-SYPSWD>>
                                        .BYTE   0
                                        .ENDR
                                        .ENDC   ;GT,<21.-<.-SYPSWD>>
                                        .EVEN
                                        .ENDM   SYSPS

                                ;-------------------------------------------------------------------------------
                                ;  The RTDEF macro is used to declare information about shared
                                ;  run-time systems.
                                ;
                                ;  The 3 arguments to RTDEF are
                                ;  1.  12 character name of run-time system file.
                                ;  2.  R or RW indicating Read-only or Read-Write access.
                                ;  3.  Number of blocks to skip at the front of the file.
                                ;
                                        .MACRO  RTDEF   NAME,RFLAG,SKIP
                                        .CSECT  RDBSEC
                                T       =       .
                                        .RAD50  /'NAME'/
                                        .IF     NE,<<.-T>-8.>
                                .ERROR  0;Run-time system name was not correctly specified
                                        .ENDC
                                        .WORD   0,0
                                        .IF     IDN,RFLAG,RW
                                        .BYTE   RF$WRT
                                        .IFF
                                        .BYTE   0
                                        .ENDC
                                        .BYTE   SKIP
                                        .CSECT  TSGEN
                                NUMRDB  =       NUMRDB+1
                                        .ENDM   RTDEF
```

```
                        ;
                        ;-------------------------------------------------
                        ;  The DHDEF macro is used to declare the beginning
                        ;  of a block of lines which are attached to a DH11
                        ;  multiplexer.  All line definition blocks up to
                        ;  the next MUXEND macro call will be connected to
                        ;  the DH11.
                        ;  There are four arguments to DHDEF:
                        ;  1. The interrupt vector address of the mux receiver.
                        ;  2. The address of the mux control and status register.
                        ;  3. The interrupt vector address of the associated DM11.
                        ;  4. The CSR address of the associated DM11.
                        ;
                                .MACRO  DHDEF   AVEC,ACSR,ADMVEC,ADMADR
                                .IF     NE CURMX
                        .ERROR  1; Missing MUXEND macro
                                MUXEND
                                .ENDC
                        LSTMX   =       LSTMX+2
                        CURMX   =       LSTMX
                        CURCDX  =       CDX$DH          ;Lines within this block are connected to DH11
                        DHUSE   =       1               ;Set flag saying DH11 support is needed
                                VECCHK  AVEC,7
                                SRCHK   ACSR
                                .IF     NE,ADMVEC
                                VECCHK  ADMVEC,3
                                .ENDC
                                .IF     NE,ADMADR
                                SRCHK   ADMADR
                                .ENDC
                        S       =       .
                        .       =       MXTYPE+CURMX    ;Type of multiplexor
                                .WORD   CDX$DH          ;Type = DH11
                        .       =       MH$SCR+CURMX    ;Status and control register address
                                .WORD   ACSR
                        .       =       MH$RCR+CURMX    ;Received character register
                                .WORD   ACSR+2
                        .       =       MH$LPR+CURMX    ;Line parameter register
                                .WORD   ACSR+4
                        .       =       MH$CAR+CURMX    ;Current address register
                                .WORD   ACSR+6
                        .       =       MH$BCR+CURMX    ;Byte count register
                                .WORD   ACSR+10
                        .       =       MH$BAR+CURMX    ;Buffer active register
                                .WORD   ACSR+12
                        .       =       MH$BRK+CURMX    ;Break control register
                                .WORD   ACSR+14
                        .       =       MH$SSR+CURMX    ;Silo status register
                                .WORD   ACSR+16
                        .       =       DM$CSR+CURMX    ;DM11 Control Status register
                                .WORD   ADMADR
                        .       =       DM$LSR+CURMX    ;DM11 Line status register
                                .WORD   ADMADR+2
                        .       =       MXVEC+CURMX     ;DH11 Interrupt vector address
                                .WORD   AVEC
                        .       =       DM$VEC+CURMX    ;DM11 Interrupt vector address
                                .WORD   ADMVEC
```

```
58                                      =       S
59                                      .ENDM   DHDEF
```

```
                        ;
                        ;-----------------------------------------------------
                        ;  The DHVDEF macro is used to declare the beginning
                        ;  of a block of lines which are attached to a DHV11
                        ;  multiplexer.  All line definition blocks up to
                        ;  the next MUXEND macro call will be connected to
                        ;  the DHV11.
                        ;  There are two arguments to DHVDEF:
                        ;  1. The interrupt vector address of the mux receiver.
                        ;  2. The address of the mux control and status register.
                        ;
                                .MACRO  DHVDEF  AVEC,ACSR
                                .IF     NE CURMX
                        .ERROR  1; Missing MUXEND macro
                                MUXEND
                                .ENDC
                        LSTMX   =       LSTMX+2
                        CURMX   =       LSTMX
                        CURCDX  =       CDX$VH          ;Lines within this block connected to DHV11
                        DHUSE   =       1               ;Set flag saying DHV11 support is needed
                                VECCHK  AVEC,7
                                SRCHK   ACSR
                        S       =       .
                        .       =       MXTYPE+CURMX    ;Type of multiplexor
                                .WORD   CDX$VH          ;Type = DHV11
                        .       =       VH$CSR+CURMX    ;Status and control register address
                                .WORD   ACSR
                        .       =       VH$DBR+CURMX    ;Data buffer register
                                .WORD   ACSR+2
                        .       =       VH$LPR+CURMX    ;Line parameter register
                                .WORD   ACSR+4
                        .       =       VH$LSR+CURMX    ;Line Status Register
                                .WORD   ACSR+6
                        .       =       VH$LCR+CURMX    ;Line Control Register
                                .WORD   ACSR+10
                        .       =       VH$BA1+CURMX    ;Buffer Address register 1
                                .WORD   ACSR+12
                        .       =       VH$BA2+CURMX    ;Buffer Address register 2
                                .WORD   ACSR+14
                        .       =       VH$BCR+CURMX    ;Byte Count Register
                                .WORD   ACSR+16
                        .       =       MXVEC+CURMX     ;Interrupt vector address
                                .WORD   AVEC
                        .       =       S
                                .ENDM   DHVDEF
                        ;
                        ;-----------------------------------------------------
                        ;  The DHUDEF macro is used to declare the beginning
                        ;  of a block of lines which are attached to a DHU11
                        ;  multiplexer.  All line definition blocks up to
                        ;  the next MUXEND macro call will be connected to
                        ;  the DHU11.
                        ;  There are two arguments to DHUDEF:
                        ;  1. The interrupt vector address of the mux receiver.
                        ;  2. The address of the mux control and status register.
                        ;
                                .MACRO  DHUDEF  AVEC,ACSR
```

```
58                                      DHVDEF  AVEC,ACSR       ;Handle same as DHV11
59                                      .ENDM   DHUDEF
```

```
                        ;
                        ;-------------------------------------------------------
                        ;  The MUXDEF macro is used to declare the beginning
                        ;  of a block of lines which are attached to a DZ11
                        ;  multiplexer.  All line definition blocks up to
                        ;  the next MUXEND macro call will be connected to
                        ;  the DZ11.
                        ;  There are two arguments to MUXDEF:
                        ;  1. The interrupt vector address of the mux receiver.
                        ;  2. The address of the mux control and status register.
                        ;
                                .MACRO  MUXDEF  AVEC,ACSR
                                .IF     NE CURMX
                        .ERROR  1; Missing MUXEND macro
                                MUXEND
                                .ENDC
                        LSTMX   =       LSTMX+2
                        CURMX   =       LSTMX
                        CURCDX  =       CDX$DZ          ;Lines within this block are connected to DZ11
                                VECCHK  AVEC,7
                                SRCHK   ACSR
                        S       =       .
                        .       =       MXTYPE+CURMX    ;Multiplexor type
                                .WORD   CDX$DZ          ;Type = DZ11
                        .       =       MXCSR+CURMX
                                .WORD   ACSR
                        .       =       MXLPR+CURMX
                                .WORD   ACSR+2
                        .       =       MXTCR+CURMX
                                .WORD   ACSR+4
                        .       =       MXDTR+CURMX
                                .WORD   ACSR+5
                        .       =       MXTBUF+CURMX
                                .WORD   ACSR+6
                        .       =       MXCAR+CURMX
                                .WORD   ACSR+7
                        .       =       MXVEC+CURMX
                                .WORD   AVEC
                        .       =       S
                                .ENDM   MUXDEF
                        ;
                        ;  Alternate name for MUXDEF macro
                        ;
                                .MACRO  DZDEF   AVEC,ACSR
                                MUXDEF  AVEC,ACSR
                                .ENDM   DZDEF
                        ;
                        ;-------------------------------------------------------
                        ;  The MUXEND macro is called to declare the end of
                        ;  a set of lines connected to a DZ11 or DH11.
                        ;
                                .MACRO  MUXEND
                                .IF     EQ CURMX
                        .ERROR  6; Missing earlier MUXDEF
                                .ENDC
                        CURMX   =       0
                        CURCDX  =       CDX$DL          ;Following lines are connected to DL11's
```

```
      58                                                .ENDM    MUXEND
```

```
                        ;-----------------------------------------------------
                        ;  The SPOOL macro is used to declare those devices which
                        ;  are to be spooled by TSX-Plus (such as line printers).
                        ;  There are seven arguments to spool:
                        ;    1) Number of devices to be spooled (may be zero)
                        ;    2) Number of spool files allowed to be open.
                        ;    3) Number of buffers for spooler to use.
                        ;    4) Number of blocks in spool disk file.
                        ;    5) List of 3 character names of devices to be spooled.
                        ;    6) 0 for 'nohold' mode, 1 for 'hold' mode.
                        ;    7) # of blocks which will be remembered for back up.
                        ;
                            .MACRO  SPOOL   SND,SNF,SNB,SNDB,SNAM,SHLD,SNBU
                        ;  Define number of spooled devices
                        SPLND   =       SND
                        SNBUX   =       SNBU
                            .IF     EQ,SNBU
                        SNBUX   =       1
                            .ENDC
                        PVSPBL  =       SNBUX+10.               ;# PRIVATE BLOCKS PER DEV
                        SNDBX   =       SNDB            ;TOTAL # OF SPOOL BLOCKS
                            .IF     LT <SNDB-<SND*PVSPBL>-2>
                        SNDBX   =       <SND*PVSPBL>+2
                            .ENDC
                            .IF     GT,SPLND
                        ;**
                        ;**  Assemble this code if there are spooled devices.
                        ;**
                        SPLNF   =       SNF             ;DEFINE # OF SPOOL FILES
                        SPLNB   =       SNB             ;DEFINE # OF SPOOL BUFFERS
                        NESB:   .WORD   SPLNB
                        ;   DEFINE SPOOL BUFFERS
                            .IF     EQ,SPLNB            ;THERE MUST BE AT LEAST 1 BUFFER
                        .ERROR  ;There must be at least 1 buffer for spooler
                        SPLNB   =       1               ;FORCE 1 BUFFER
                            .ENDC
                        SPLBHD: .WORD   0               ;HEAD OF FREE BUFFER CHAIN
                            .IF     EQ,SPLNF            ;Make sure we have at least 1 file
                        .ERROR  ;There must be at least 1 spool file
                        SPLNF   =       1               ;FORCE 1 FILE
                            .ENDC
                        ;
                        ;  Define spool device control blocks (SDCB)
                        ;
                        SDCB:
                            .REPT   SPLND
                            .WORD   0,0,0,0,0           ;SDCHAN
                            .WORD   0,0,0,0,0,0
                            .WORD   0,0,0,0
                        ;   INITIAL FORM NAME
                            .ASCII  /STD    /           ;SDFORM
                            .WORD   0,0                 ;SDANAM
                        ;   GEN INIT FLAGS
                            .IF     NE,SHLD
                            .WORD   SD$HLD              ;SDFLAG
                            .IFF
                            .WORD   0                   ;SDFLAG
```

```
                        .ENDC
                        .WORD   0                       ;SDSKIP
                        .WORD   PVSPBL                  ;SDFRBL
                ;   GEN BACKUP CELLS
                        .REPT   SNBUX
                        .WORD   0
                        .ENDR
                        .WORD   0                       ;SDBULS = END OF SDBU
                        .ENDR
                SDCBND:                                 ;END OF SDCB AREA
                ;   DEFINE SIZE OF SDCB
                SDCBSZ  =       48.+<2*SNBUX>
                SDBULS  =       SDCBSZ-4.
                ;
                ;   Define table of device names.
                ;
                SPLDEV: .RAD50  /'SNAM'/                ;DEFINE TABLE OF NAMES
                        .EVEN
                SPLDVN:                                 ;END OF TABLE
                        .IF     NE,<<SPLDVN-SPLDEV>-<2*SND>>
                .ERROR  ;Number of spooled devices not equal to number of names
                        .ENDC
                SPLANM: .ASCII  /'SNAM'/
                        .EVEN
                ;   Reserve space for spool file channel block
                SPLCHN: .BLKW   5
                ;
                        .IFF
                ;**
                ;**  This code is assembled if there are no spooled devices.
                ;**
                SPLND   =       0
                SPLNF   =       0
                SPLNB   =       0
                SDCBSZ  =       1
                SDBULS  =       1
                ;
                SPLDEV:
                SPLDVN:
                SPLANM:
                SPLBHD:
                SPLCHN:
                SDCB:
                SDCBND:
                NESB:
                        .WORD   0                       ;SAY ALL LISTS ARE EMPTY
                ;
                        .ENDC
                        .ENDM   SPOOL
```

```
                                        ;================================================================
                                        ;  The TSX-Plus system manager alters values in the following
                                        ;  section to customize the system for a particular configuration.
                                        ;
                                        ;  System parameters:
                                        ;
                                        ;  Swap file device-file specification (do not place on VM).
                                        ;
003102  075250  100020  075150  SWDBLK: .RAD50  /SY TSXSWPTSX/
003110  100020
                                        ;
                                        ;  Spool file device-file specification (do not place on VM).
                                        ;
003112  075250  100020  074514  SPLBLK: .RAD50  /SY TSXSPLTSX/
003120  100020
                                        ;
                                        ;  PLAS region swap file specification (do not place on VM).
                                        ;
003122  075250  100020  071576  RSFBLK: .RAD50  /SY TSXRSFTSX/
003130  100020
                                        ;
                                        ;  File spec for file used to hold user defined command definitions (UCL)
                                        ;
003132  075250  100020  101704  UCLDAT: .RAD50  /SY TSXUCLTSX/
003140  100020
                                        ;
                                        ;  File spec for temp file used while processing IND command files
                                        ;
003142  075250  100020  035164  INDFIL: .RAD50  /SY TSXINDTSX/
003150  100020
                                        ;
                                        ;  Maximum amount of memory that can be used by any job (# K bytes).
                                        ;  This value must not exceed 64. (Kb)
                                        ;
        000100                          HIMEM   =       64.     ;Max memory that any job may use
                                        ;
                                        ;  Default memory size for jobs that will be in effect when the job
                                        ;  logs on.  (Specify in # K bytes).
                                        ;
        000070                          DFLMEM  =       56.     ;Default memory limit for jobs
                                        ;
                                        ;  SWAPFL controls whether TSX-Plus is allowed to swap jobs to disk if
                                        ;  insufficient memory is available to hold all active users.
                                        ;  The normal case (SWAPFL=1) allows TSX-Plus to do job swapping.
                                        ;  SWAPFL can be set to 0 (zero) in special situations such as when a
                                        ;  small number of lines are being supported on a floppy disk based system
                                        ;  that does not have room for a swap file.
                                        ;  If SWAPFL is set to zero the following actions occur:
                                        ;  1. No disk swap file is created.
                                        ;  2. A line will not be allowed to log on if there is insufficient
                                        ;     free memory space to support it.
                                        ;  3. Each job is allocated a memory size equal to DFLMEM (default job
                                        ;     memory size).
                                        ;  4. The MEMORY command cannot be used to change the job size.
                                        ;
        000001                          SWAPFL  =       1       ;1==>Allow job swapping; 0==>Do not swap.
                                        ;
```

```
                                ;  If the system is generated with job swapping enabled (SWAPFL=1), then
                                ;  the SWPSLT parameter controls the number of job slots allocated
                                ;  in the swap file.  SWPSLT should be in the range 0 up to the
                                ;  total number of jobs. If SWPSLT is set to zero, TSX-Plus will
                                ;  automatically allocate one job slot in the swap file for each job.
                                ;  SWPSLT may be set to a value less than the total number of jobs if
                                ;  a small amount of job swapping is anticipated; however, a system
                                ;  crash will occur if the system needs to swap a job out of memory
                                ;  and no free slot is available in the swap file.
                                ;  The SWPSLT parameter has no effect on non-swapping systems (SWAPFL=0).
                                ;  The recommended setting for this parameter is 0 (zero).
                                ;
000000                          SWPSLT  =       0.      ;Number of job slots in swap file
                                ;
                                ;  Number of 512-byte blocks to allocate for swap file that is used
                                ;  for extended memory PLAS (Program's Logical Address Space) regions
                                ;  that are used by jobs that have virtual overlays or virtual arrays.
                                ;  Note that this is the total space in the PLAS swap file for all
                                ;  extended memory regions in use at any time by all jobs.
                                ;  Note: In a non-swapping system (SWAPFL=0), SEGBLK must be non-zero
                                ;  if PLAS support is wanted, but its value does not matter.
                                ;
000000                          SEGBLK  =       0.      ;# blocks for PLAS swap file
                                ;
                                ;  Number of shared global PLAS regions that can be created by all jobs.
                                ;
000014                          NGR     =       12.     ;Number of global PLAS regions
                                ;
                                ;  BUSTYP defines the machine bus structure for TSX-Plus.  Their are two
                                ;  possible machine bus structures supported by TSX-Plus - the QBUS (LSI)
                                ;  and the UNIBUS.  Select one of these parameters below to specify the
                                ;  bus support desired.  Use the following information for choosing the
                                ;  correct bus structure.
                                ;
                                ;  QBUS   - 11/23, 11/23-Plus, 11/73, and Professional.
                                ;  UNIBUS - 11/24, 11/34a, 11/44, and 11/60.
                                ;
000001                          BUSTYP  =       QBUS    ;Specify machine bus structure (UNIBUS/QBUS)
                                ;
                                ;  Memory upper limit size specification expressed in number of k-bytes.
                                ;  This parameter controls the maximum memory available for TSX-Plus
                                ;  system use.  Memory above this upper limit will not be used by the
                                ;  operating system.
                                ;  If the MEMSIZ parameter is set to 0 (zero), TSX-Plus will use all
                                ;  available memory on the machine.  To disable the use of extended
                                ;  memory, set MEMSIZ to 248 or less.
                                ;
000370                          MEMSIZ  =       248.    ;Upper memory limit
                                ;
                                ;  The INIABT parameter controls the action taken by TSX-Plus when
                                ;  certain errors are detected during system initialization.
                                ;  If INIABT=0, TSX-Plus ignores the error and continues running.
                                ;  If INIABT=1, TSX-Plus aborts initialization and prints an error message.
                                ;
                                ;  ********************************************
                                ;  **  The normal and recommended setting for  **
                                ;  **  this parameter is INIABT=1. It is cleared  **
```

```
                                ;  **  for default installation.                    **
                                ;  ************************************************
                                ;
                                ;  The following initialization errors are controlled by the INIABT flag:
                                ;    1. A device that was specified in TSGEN does not have a
                                ;       TSX-Plus handler on the system disk.
                                ;    2. A time sharing line that was generated into TSX-Plus is not
                                ;       installed on the machine.
                                ;    3. A shared run-time system file could not be found during startup.
                                ;
000000                          INIABT  =       0       ;0==>Continue on error, 1==>Abort on error
                                ;
                                ;  The UXIFLG parameter controls the action taken by TSX-Plus when
                                ;  an interrupt occurs at an unexpected location. Unexpected interrupts
                                ;  may occur if the interrupt vector address specified in a device
                                ;  handler does not match the actual interrupt address for which the
                                ;  device has been set. Unexpected interrupts can also occur if real-time
                                ;  interrupts occur and no connection has been established between the
                                ;  real-time interrupt and a TSX-Plus real-time program.
                                ;
                                ;  If UXIFLG is set to 1 (one) then unexpected interrupts cause a system
                                ;  crash with the error message:
                                ;       ?TSX-F-UEI-Interrupt occurred at unexpected location
                                ;       Argument value = xxxx
                                ;  Where "xxxx" is the address at which the interrupt occurred.
                                ;
                                ;  If UXIFLG is set to 0 (zero) then unexpected interrupts are ignored
                                ;  by the system and do not cause a crash or print an error message.
                                ;
                                ;  The recommended setting for UXIFLG is 1 (one).
                                ;
000001                          UXIFLG  =       1       ;Unexpected interrupt control flag
                                ;
                                ;  Parameters related to the TSX-Plus system crash dump facility.
                                ;  This optional facility will print some useful internal system
                                ;  data if a system crash occurs.  The dump information can be printed
                                ;  on any terminal connected to a DL-11 type line (including DLV-11)
                                ;  or on a parallel printer port.
                                ;  It is recommended that this facility not be included in the system
                                ;  unless you are experiencing system crashes.
                                ;
                                ;  Set SYSDMP to 1 if you want the crash dump facility, 0 if not.
                                ;
000000                          SYSDMP  =       0       ;1==>Enable crash dump, 0==>No crash dump
                                ;
                                ;  Address of transmitter control register for device to which crash
                                ;  dump is to be written.  This must be a DL-11 type device controller
                                ;  or a parallel printer controller.
                                ;  Specify 177564 to dump on the console terminal.
                                ;  Specify 177514 to dump to line printer connected to standard parallel port.
                                ;
177560                          DMPTCR  =       177560  ;Transmitter control reg for dump device
                                ;
                                ;  Set DMPKTP to 1 if you want a system crash to occur any time a trap
                                ;  occurs within the system. Set it to 0 (zero) if you want recoverable
                                ;  traps within the system to abort the job but continue execution of the
                                ;  system.
```

```
                                ;
000000                          DMPKTP  =       0       ;1==>Always crash on traps within system
                                ;
                                ;  The  IOABT parameter controls the action taken by TSX-Plus when
                                ;  a job terminates execution.  If IOABT=0, TSX-Plus will wait for
                                ;  all outstanding I/O pending for the job to complete before the job
                                ;  is actually terminated.  If IOABT=1, TSX-Plus will call the handler
                                ;  abort entry point for all outstanding I/O pending for the job.
                                ;  Note, the "SET IO [NO] ABORT" keyboard command may be used to
                                ;  change the value of this parameter.
                                ;
000001                          IOABT   =       1       ;0==>I/O rundown, 1==>I/O abort
                                ;
                                ;  U$CL is a flag that controls whether the User Command Linkage is to
                                ;  be used to allow users to define their own commands.
                                ;  If U$CL is non-zero the UCL facility is enabled and users may define
                                ;  their own system commands. If U$CL is zero, user defined commands
                                ;  will not be supported by the system. Note: if the UCL facility is
                                ;  enabled, the TSXUCL.SAV file must be placed on the system disk.
                                ;
000001                          U$CL    =       1       ;0==>No UCL program, 1==>UCL program
                                ;
                                ;  Number of user-defined commands that can be stored by TSXUCL
                                ;  for each job. (The number of blocks required in the SY:TSXUCL.DAT file
                                ;  is approximately equal to the number of commands per job times the
                                ;  total number of time-sharing lines divided by 5).
                                ;
000005                          UCLMNC  =       5.      ;Maximum user-defined commands per job
                                ;
                                ;  The UCLORD parameter selects the default call order for checking
                                ;  to see if a command is a user-defined command.
                                ;  FIRST  ==> Check for user-defined commands before system commands.
                                ;  MIDDLE ==> Check after system commands but before command files.
                                ;  LAST   ==> Check after system commands and command files.
                                ;
                                ;  Note that the SET UCL FIRST/LAST keyboard command can be used to
                                ;  alter this order on a line-by-line basis.
                                ;
000002                          UCLORD  =       MIDDLE  ;Select FIRST / MIDDLE / LAST
                                ;
                                ;  The LDSYS flag controls whether the standard system support for
                                ;  logical disks (LD) is to be provided.
                                ;  If LDSYS is set to 1, system support for logical disks is included.
                                ;  If LDSYS is set to 0, system support for logical disks is excluded.
                                ;
000001                          LDSYS   =       1       ;1==>Include LD support, 0==>Exclude LD.
                                ;
                                ;  The SLEDIT flag controls whether the Single Line Editor (SL) facility
                                ;  is to be made available to the system.
                                ;  If SLEDIT is set to 1, Single Line Editor support is included.
                                ;  If SLEDIT is set to 0, Single Line Editor support is omitted.
                                ;  Single Line Editor support adds approximately 2Kb to the size of the
                                ;  mapped portion of the system.
                                ;
000001                          SLEDIT  =       1       ;1==>Include SL support, 0==>Exclude SL
                                ;
                                ;  The KEYMAX parameter specifies the number of user-defined keys supported
```

```
                                ;  by the single line editor.  The DEFINE/KEY command is used to associate
                                ;  a user-specified text string with a function key.  The maximum number
                                ;  of such key definitions that may be in effect at one time for each user
                                ;  is controlled by the KEYMAX parameter.
                                ;  The maximum supported value for KEYMAX is 60.
                                ;
000007                          KEYMAX  =       7.      ;Maximum number of user-defined keys for SL
                                ;
                                ;  The MAXWIN parameter specifies the maximum number of terminal display
                                ;  windows that may be in use by all jobs on the system.
                                ;  If MAXWIN is set to 0 (zero), the display window feature is not included
                                ;  in the system.  Display windows are useful if you frequently utilize
                                ;  subprocesses in that they preserve the screen context when you switch
                                ;  between processes.
                                ;
000012                          MAXWIN  =       10.     ;Total number of display windows for all jobs
                                ;
                                ;  Set DBGFLG to 1 to cause the TSX-Plus program debugging facility
                                ;  to be included with the system.
                                ;  Set DBGFLG to 0 if the debugging facility is not wanted.
                                ;
000000                          DBGFLG  =       0       ;1==>Include debugger; 0==>Exclude debugger
                                ;
                                ;  Number of slots in INSTALL table to reserve for user programs.
                                ;
000004                          NUIP    =       4.      ;Number of INSTALL slots for user programs
                                ;
                                ;  The following time-slice values are used to schedule jobs for execution.
                                ;  Each time value must be specified in 0.1 second units.
                                ;
                                ;  QUAN0 -- Time slice for round-robin scheduling of high-priority
                                ;           real-time jobs. That is, jobs with execution priorities
                                ;           greater than or equal to PRIHI.
                                ;
000002                          QUAN0   =       2.      ;Time slice for real-time jobs
                                ;
                                ;  QUAN1 -- Time that jobs will remain in a high-priority state after
                                ;           they receive an activation character from the terminal.
                                ;           A job is classified as "interactive" from the time when an
                                ;           activation character is received until the job consumes
                                ;           QUAN1 units of time, then the job is classified as "compute
                                ;           bound".
                                ;
000024                          QUAN1   =       20.     ;High-priority time for interactive jobs
                                ;
                                ;  QUAN1A -- Time that jobs will remain in a high-priority state after
                                ;            they are activated because of I/O completion or they are
                                ;            restarted following other wait states.
                                ;
000002                          QUAN1A  =       2.      ;High-priority time for wait-reactivation
                                ;
                                ;  QUAN1B -- Time slice used to switch between "interactive" jobs.
                                ;
000002                          QUAN1B  =       2.      ;Time slice for "interactive" jobs.
                                ;
                                ;  QUAN1C -- Time job will be allowed to stay in highest execution state
                                ;            after receipt of a character from the terminal.
```

```
                                ;
000001                          QUAN1C  =       1       ;Time at highest execution state
                                ;
                                ;  QUAN2 -- Time that normal priority CPU-bound jobs are allowed to run
                                ;           if there are no high-priority jobs that want to run.
                                ;           This time-slice controls round-robin scheduling of CPU-bound jobs
                                ;           with execution priority values in the range (PRILOW+1) to
                                ;           (PRIHI-1).
                                ;
000012                          QUAN2   =       10.     ;Normal-priority CPU-bound job time-slice
                                ;
                                ;  QUAN3  -- Time slice for round-robin scheduling of very low priority
                                ;            jobs. That is, jobs with priorities less than or equal
                                ;            to PRILOW.
                                ;
000024                          QUAN3   =       20.     ;Time slice for very low priority jobs
                                ;
                                ;  INTIOC -- Number of consecutive times that a job will be allowed to
                                ;            perform I/O operations following input of an activation
                                ;            character from the terminal before the job is classified
                                ;            as non-interactive.
                                ;
000036                          INTIOC  =       30.     ;Number of I/O ops. while "interactive".
                                ;
                                ;  HIPRCT -- Number of consecutive times that a job will be given a
                                ;            high-priority execution boost following wait states such
                                ;            as I/O wait before the job will be scheduled as a normal
                                ;            CPU-bound job.
                                ;
000050                          HIPRCT  =       40.     ;Number of consecutive high-priority hits
                                ;
                                ;  Time that job will be held in memory after being swapped in from disk.
                                ;  A job is not eligible to be swapped out of memory until CORTIM has
                                ;  elapsed since it was swapped into memory.  However, the job becomes
                                ;  immediately eligible to be swapped if it goes into a state where it is
                                ;  waiting on any resource other than non-terminal I/O.
                                ;  Specify in 0.1 second units.
                                ;
000002                          CORTIM  =       2.      ;Guaranteed memory-residency time
                                ;
                                ;  Job priority classes:  There are three groups of job priorities,
                                ;  the lowest priority group ranges from a job priority 0 up to and
                                ;  including the priority equal to the PRILOW parameter. Jobs with
                                ;  priorities in this range execute with lower priority than all normal
                                ;  time-sharing jobs.
                                ;  The second range of priorities is from (PRILOW+1) up to (PRIHI-1).
                                ;  Jobs in this range are treated as normal time-sharing jobs.
                                ;  The third range of priorities is from PRIHI up to 127. These priorities
                                ;  are for real-time jobs which will take unconditional precedence over
                                ;  all other jobs.
                                ;  All priority values must be in the range 0 to 127.
                                ;
000023                          PRILOW  =       19.     ;Highest "low priority" value
000120                          PRIHI   =       80.     ;Lowest "high priority" value
                                ;
                                ;  PRIDEF -- Default job priority.
                                ;
```

```
000062                  PRIDEF  =       50.     ;Default job priority
                        ;
                        ;  PRIVIR -- Amount by which a job's execution priority is reduced
                        ;            when the job is disconnected from the terminal by switching
                        ;            to a subprocess. Note: this only applies to jobs with
                        ;            base priorities in the range (PRILOW+1) to (PRIHI-1).
                        ;
000012                  PRIVIR  =       10.     ;Disconnect job priority reduction
                        ;
                        ;  Maximum number of subprocesses per primary process.
                        ;
000002                  MAXSEC  =       2.      ;Max subprocesses per user
                        ;
                        ;  Maximum file size (# blocks) that will be returned in response to
                        ;  a .ENTER request that specifies a file size of 0 blocks.
                        ;
001750                  MAXFIL  =       1000.   ;Max # blocks for default allocation
                        ;
                        ;  Number of 512 byte blocks to hold in memory in a generalized data cache.
                        ;  If the CACHE parameter is set to 0 (zero), data caching is not performed.
                        ;  Note: The data caching facility adds approximately 2000 bytes to the
                        ;  size of the unmapped portion of the system and 528*CACHE bytes to
                        ;  the mapped portion of the system.
                        ;  The maximum number of blocks that may be held in the cache is 4095. (2MB)
                        ;
000000                  CACHE   =       0.      ;Number of blocks in data cache
                        ;
                        ;  The following parameters relate to the cache of file directory entries
                        ;  maintained by TSX-Plus.  This cache is used to reduce the number of disk
                        ;  accesses required to do lookups on frequently accessed files.
                        ;  The system disk (SY:) is automatically cached.
                        ;  Other devices are only cached if they are introduced to the system
                        ;  by use of the MOUNT command.
                        ;
                        ;  Maximum number of units that may be cached.
                        ;  This includes all logical disks (LD) and all physical disks for which
                        ;  directory caching is enabled by use of the MOUNT command.
                        ;  (Space required is 18 bytes per unit).
                        ;
000012                  MAXCSH  =       10.     ;Max # device units whose directories to cache
                        ;
                        ;  Maximum number of file entries to be held in directory cache.
                        ;  (Space required is 18 bytes per entry)
                        ;
000050                  NMFCSH  =       40.     ;Max # file entries to be cached
                        ;
                        ;  Maximum number of device units that can be allocated to jobs for exclusive
                        ;  use by use of the ALLOCATE command.
                        ;
000005                  MAXALC  =       5.      ;Max # units that can be allocated
                        ;
                        ;  Maximum number of simultaneous requests by jobs to monitor other jobs.
                        ;
000005                  MAXMON  =       5.      ;Max # job monitoring requests
                        ;
                        ;  The system password is a global password which must be entered
                        ;  when a line is initiated before the normal logon sequence begins.
```

```
                                ;  The use of a system password is optional and may be enabled on a
                                ;  line-by-line basis by specifying the $SYSPS flag with the
                                ;  FLAGS macro within the line definition blocks for the lines
                                ;  for which the password will be required.  If a system password is
                                ;  required for a line, an exclamation point prompt is printed as the
                                ;  first thing when the line is initiated.  The idea is to force the
                                ;  calling person to provide a password before printing the normal
                                ;  logon greeting which identifies the nature and identity of the site.
                                ;
003152                                SYSPS   <TSX>   ;System password for all lines with $SYSPS
                                ;
                                ;  Amount of time that carrier signal must be lost on dial-up
                                ;  lines before we assume the connection has been broken.
                                ;  This value is also used to time-out lines which ring and
                                ;  do not raise carrier.
                                ;  Specify in 0.5 second units.
                                ;
        000036                  TIMOUT  =       30.     ;Time allowed for lost carrier
                                ;
                                ;  Amount of time that a user may remain connected to a dial-up line
                                ;  after logging off before Data Terminal Ready (DTR) will be
                                ;  dropped causing the phone to hang up.
                                ;  Specify in 0.5 second units.
                                ;
        000074                  OFFTIM  =       60.     ;Time allowed for job to be logged off
                                ;
                                ;  Modem lines ($PHONE in the LINDEF FLAGS macro) are normally
                                ;  treated as phone lines if the DCD signal (carrier) is present
                                ;  when the lines are started and optionally treated as local lines
                                ;  if the signal is not present. The OFFTIM and TIMOUT parameters
                                ;  are only effective if the line is recognized as a phone line when
                                ;  started. Set PHONE to 0 to allow lines with the $PHONE
                                ;  flag to optionally support local lines. Set PHONE to 1 to
                                ;  force the OFFTIM and TIMOUT parameters to always take effect for
                                ;  lines with the $PHONE flag.
                                ;
000000                          PHONE   =       0.      ;$PHONE lines may be local if carrier absent
                                ;
                                ;  Define Lead-in character that tells TSX-Plus that a special
                                ;  terminal control sequence is coming from the program.
                                ;
000035                          TSLICH  =       035     ;Octal 35 = decimal 29.
                                ;
                                ;  Define the keyboard control character that will be used to
                                ;  switch to a subprocess.
                                ;  (Specify the octal value of the ASCII control character)
                                ;
000027                          VLSWCH  =       027     ;Octal 27 = control-W
                                ;
                                ;  Define keyboard control character used to cause the current screen
                                ;  window contents to be printed.
                                ;  (Specify the octal value of the ASCII control character)
                                ;
000002                          PWCH    =       002     ;Octal 02 = control-B
                                ;
                                ;  Define keyboard control character that is used to terminate
                                ;  a cross-connection between a time-sharing line and a CL line.
```

```
                                ;  (Specify the octal value of the ASCII control character)
                                ;
000034                          CCXTRM  =       034     ;Octal 34 = control-\ (control backslash)
                                ;
                                ;  Define keyboard control character that is used to signal
                                ;  special control functions for a time-sharing line cross-connected
                                ;  to a CL line.
                                ;  (Specify the octal value of the ASCII control character)
                                ;
000001                          CCXCTL  =       001     ;Octal 001 = control-A
                                ;
                                ;  Define the version number to be associated with the CL handler when
                                ;  being used with VTCOM.  If CLVRSN is defined as 0 then an appropraite
                                ;  value will be selected via an internal table. Zero is the suggested
                                ;  setting.
                                ;
000000                          CLVRSN  =       0.      ;CL version number
                                ;
                                ;  Define maximum number of user defined activation characters
                                ;  that each line may define during execution.
                                ;
000020                          MXSPAC  =       16.     ;Max # user defined activation chars per job
                                ;
                                ;  Define maximum number of characters that can be translated by
                                ;  the terminal handler.  This translation consists of replacing
                                ;  a received character by a substitution character on input and replacing
                                ;  the substitution character by the original character on output.
                                ;  This parameter must be non-zero to use the SET TT TRANSLATE=( ) command.
                                ;
000004                          MXTTCT  =       4.      ;Max # chars that terminal handler can translate
                                ;
                                ;  Select default system editor.
                                ;  The choices are
                                ;   EDIT
                                ;   TECO
                                ;   KED
                                ;   K52
                                ;
000003                          EDITOR  =       KED     ;Default system editor
                                ;
                                ;  Select system default implicit or explicit wildcards for CCL commands.
                                ;    If WILDFL = 0 then explicit wildcards are selected.
                                ;    If WILDFL = 1 then implicit wildcards are selected.
                                ;
000001                          WILDFL  =       1       ;1==>Implicit wildcard, 0==>Explicit wildcard
                                ;
                                ;-----------------------------------------------------------------------
                                ;  The DEVDEF macro must be used to define the names and characteristics
                                ;  of all devices which are to be available to TSX-Plus users.
                                ;  The form of a device definition is:
                                ;
                                ;       DEVDEF  <device>[,option,...,option]
                                ;
                                ;  For each device to be available to the system an entry must be made
                                ;  using the DEVDEF macro. This macro requires at least one argument
                                ;  but may have several optional arguments as described below:
                                ;
```

```
                                ;  1.   The first parameter is the two character device name enclosed
                                ;       in angle brackets.
                                ;  2.   The optional parameters specify the device characteristics.
                                ;       There are nine allowable device attributes which may be
                                ;       specified in any order.  They are as follows:
                                ;
                                ;       DMA       Device performs Direct Memory Access (DMA).
                                ;       MAPIO     Perform I/O mapping (18-bit controllers on 22-bit QBUS).
                                ;       EVNBUF    Require even byte buffer address for I/O transfers.
                                ;       NOCACHE   Do not use generalized data cache for this device.
                                ;       NOMOUNT   Do not allow mounts (i.e., use directory cache) for
                                ;                 this device.
                                ;       REQALC    Require device allocation before use.
                                ;       MAPH      Load the device handler outside the low memory 40K
                                ;                 byte region and into a mapped handler region.
                                ;       NOMAPH    Do not load the handler into a mapped handler region
                                ;                 instead load it into the low memory 40k byte region.
                                ;       HANBUF    Handler contains an internal I/O buffer.
                                ;
                                ;  For standard device drivers, it is important to choose MAPIO when
                                ;  18-bit controllers or handlers will be used on a 22-bit LSI system.
                                ;  It is not necessary to specify other device attributes for standard
                                ;  TSX-Plus supplied device drivers since TSX-Plus will automatically
                                ;  make default selections.
                                ;
                                ;       ************************************
                                ;       **  When performing a TSX-Plus    **
                                ;       **  system generation, remove the  **
                                ;       **  devices in this list which are **
                                ;       **  not present on your system,    **
                                ;       **  and include those which are.   **
                                ;       ************************************
                                ;
003200                                  DEVBEG          ;Beginning of device definitions
003200                                  DEVDEF  <DL>
003200                                  DEVDEF  <DM>
003200                                  DEVDEF  <DU>,NOSET
003200                                  DEVDEF  <RK>,MAPIO
003200                                  DEVDEF  <DY>,MAPIO
003200                                  DEVDEF  <DX>
003200                                  DEVDEF  <LP>
003200                                  DEVDEF  <NL>
003200                                  DEVEND          ;End of device definitions

                                ;-------------------------------------------------------------------------
                                ;  Parameters related to system I/O buffers used when DMA devices
                                ;  with 18-bit controllers are used on Q-bus systems with
                                ;  22-bit addressing (e.g., 11/23-Plus and 11/73).
                                ;
                                ;  Number of system buffers allocated for I/O buffering.
                                ;  (The recommended number is one per active device that requires buffering.)
                                ;
000001                          MIONBF  =       1.      ;Number of system I/O buffers
                                ;
                                ;  Size of each system I/O buffer, in units of 512 bytes.
                                ;  The maximum allowed value for this parameter is 15.
                                ;
```

```
                                MIOBSZ  =       8.      ;I/O buffer size in units of 512 bytes

                                ;-------------------------------------------------------------------------
                                ;  Some device handlers allocate extended memory (PLAS) regions for
                                ;  their use.  For example, the DU and MU handlers each require one
                                ;  PLAS region.  If you are using any other handlers which require
                                ;  extended memory regions, include the number of regions required.
                                ;
                                DEVXMR  =       4.      ;Number of XM regions for device handlers

                                ;-------------------------------------------------------------------------
                                ;  Define those devices which are to be spooled by TSX-Plus
                                ;  (such as line printers).
                                ;  There are seven arguments to the SPOOL macro:
                                ;    1.  Number of devices to be spooled (may be zero).
                                ;    2.  Number of spool files which may be open by all users.
                                ;    3.  Number of spool buffers (512. bytes each).
                                ;    4.  Number of blocks in spool disk file.
                                ;    5.  List of 3 character names of devices to be spooled.
                                ;    6.  Specify 0 if spool files are to be eligible to be
                                ;        started as soon as they are created,
                                ;        specify 1 if they are to be held until the channel
                                ;        is closed.  Note: The "SPOOL xx,[NO]HOLD" keyboard
                                ;        command can override this parameter.
                                ;    7.  Number of blocks which are to be backed up
                                ;        when the "SPOOL xx,BACK" command is given.
                                ;
                                ;  Note: The SPOOL macro must be present even if
                                ;  there are no spooled devices.  However, if the first
                                ;  argument (number of spooled devices) is zero, no spool
                                ;  tables are generated and arguments 2-7 are ignored.
                                ;
003200                                  SPOOL   1,20.,2,500.,<LP >,0,10.

                                ;-------------------------------------------------------------------------
                                ;  Define parameters pertaining to record (block) locking
                                ;  for shared files.  If the shared file block locking
                                ;  facility is not wanted, set all of these parameters to
                                ;  0 (zero).
                                ;
                                ;  Maximum number of shared files which may be open
                                ;  simultaneously.  Note that several users accessing the same
                                ;  file count as 1.
                                ;
        000036                  MAXSF   =       30.     ;Max number of shared files
                                ;
                                ;  Maximum number of I/O channels which all users may
                                ;  simultaneously have open to shared files.
                                ;  Note, this is the total number for all users not
                                ;  for each user.
                                ;
        000036                  MAXSFC  =       30.     ;Max # shared file channels
                                ;
                                ;  Maximum number of blocks which may be simultaneously
                                ;  held locked by any channel.  That is, max blocks
                                ;  locked per channel.
                                ;
```

```
623         000003                  MXLBLK  =       3.      ;Max blocks locked per channel
624                                 ;
625                                 ;  Number of 512-byte blocks to be held in the in-memory data
626                                 ;  cache for shared files.
627                                 ;  (Note that the MAXSF, MAXSFC, and MXLBLK parameters must be
628                                 ;  non-zero to enable shared file data caching.)
629                                 ;
630         000000                  NUMDC   =       0.      ;Number of blocks in shared file data cache
631                                 ;
632                                 ;---------------------------------------------------------------
633                                 ;  Define parameters pertaining to the inter-program
634                                 ;  message communication feature.  If this feature is
635                                 ;  not wanted, set all four parameters to 0 (zero).
636                                 ;
637                                 ;  Maximum number of message communication channels
638                                 ;  which may be simultaneously in use.
639                                 ;
640         000003                  MAXMC   =       3.      ;Max message channels
641                                 ;
642                                 ;  Maximum message length (bytes).
643                                 ;
644         000310                  MSCHRS  =       200.    ;Max message length (bytes)
645                                 ;
646                                 ;  Maximum number of messages which may be held in queue.
647                                 ;
648         000003                  MAXMSG  =       3.      ;Max queued messages
649                                 ;
650                                 ;  Maximum number of requests for messages that may be held in queue
651                                 ;
652         000012                  MAXMRB  =       10.     ;Max # pending message requests
653
654                                 ;---------------------------------------------------------------------
655                                 ;  The RTVECT parameter specifies the number of real-time interrupt vectors
656                                 ;  that can be connected to TSX-Plus jobs.  Set RTVECT to the maximum number
657                                 ;  of interrupt vectors that all running real-time programs may be connected
658                                 ;  to at the same time.
659                                 ;  (Note: The basic real-time support facility is now a standard part of
660                                 ;  TSX-Plus and it is no longer necessary to set RTVECT to 1 to include
661                                 ;  real-time facilities such as locking a job in memory or accessing the
662                                 ;  I/O page.  It is also no longer necessary to set RTVECT to 1 to allow
663                                 ;  use of the SYSMON program. RTVECT should be set to 0 (zero) unless some
664                                 ;  real-time interrupts are going to be connected to TSX-Plus jobs.)
665                                 ;
666         000000                  RTVECT  =       0.      ;Max # interrupt vectors that may be connected
667
668                                 ;-----------------------------------------------------------------
669                                 ;  Define the size of the table within TSX-Plus used to hold information
670                                 ;  when the performance monitoring feature is being used.
671                                 ;  Each word in this table corresponds to one cell in the histogram.
672                                 ;  Specify the size as number of bytes for the table.
673                                 ;  (Note: The maximum allowed size is 8192 bytes)
674                                 ;
675         000000                  PMSIZE  =       0.      ;Size of performance monitor table (bytes)
676
677                                 ;---------------------------------------------------------------------
678                                 ;  Use the RTDEF macro at this point to specify information about
679                                 ;  any shared run-time systems to be loaded when TSX-Plus is started.
```

```
                                ;
                                ;  The form of the RTDEF macro is
                                ;       RTDEF   <name>,r-flag,skip-count
                                ;
                                ;  Where
                                ; - Name is the 12 character name of the file containing the run-time system
                                ;   which must be specified in the form DevFilnamExt -- that is, three
                                ;   character device name, six character file name and three character
                                ;   extension.
                                ; - R-flag is either R if user programs are to have read-only access to
                                ;   the run-time system, or RW if read-write access is to be granted.
                                ; - Skip-count is the number of blocks to be skipped over at the front
                                ;   of the file when loading it.
                                ;
                                ;  Example:
                                ;       RTDEF   <SY CBR063SHR>,R,1.     ;COBOL-Plus shared run-time
                                ;       RTDEF   <SY DBLSHRRTS>,R,1.     ;DBL shared run-time
                                ;       RTDEF   <SY DB4RTSSHR>,R,0.     ;DBL V4 shared run-time
                                ;
```

```
                                ;-----------------------------------------------------------------------
                                ;  Time-sharing line parameters:
                                ;
                                ;  Default input and output character buffer sizes.
                                ;  These buffer sizes will be used for lines that don't use
                                ;  the BUFSIZ macro within their line definitions to declare
                                ;  their character buffer sizes.
                                ;  These buffer sizes are also used for all subprocesses.
                                ;
000144                          DINSPC  =       100.    ;Default input char buffer size
000360                          DOTSPC  =       240.    ;Default output char buffer size
                                ;
                                ;  When the terminal-output character buffer is filled a job is suspended.
                                ;  The job is restarted after characters are printed from the buffer and
                                ;  there are OTRASZ characters remaining in the buffer.
                                ;
000031                          OTRASZ  =       25.     ;Reactivation character count
                                ;
                                ;  A software character "silo" is used to hold characters received
                                ;  from time-sharing lines until they can be processed by the system.
                                ;  The silo is used to prevent the loss of characters during high
                                ;  speed input.  Each time-sharing line and CL line has its own silo.
                                ;  If the input to the line is coming from a terminal, the silo can be
                                ;  quite small.  On the other hand, if the input is coming from another
                                ;  computer or other high speed device, the silo size should be increased.
                                ;  The NCSILO, NCXOFF, and NCXON parameters set default values pertaining
                                ;  to the silos.  The SILO macro can be used within a line definition
                                ;  to specify silo parameters for a specific line.
                                ;
                                ;  Default size of input character silos.
                                ;
000040                          NCSILO  =       32.     ;Default silo size
                                ;
                                ;  The system will transmit a control-S (XOFF) character when an input
                                ;  silo is filled to the point where there are only NCXOFF free
                                ;  character positions remaining.
                                ;
000014                          NCXOFF  =       12.     ;Default XOFF point for silos
                                ;
                                ;  If the system sends an XOFF because a silo becomes nearly full,
                                ;  it will send an XON to restart transmission when there are only
                                ;  NCXON characters remaining in the silo.
                                ;
000004                          NCXON   =       4.      ;Default XON point for silos
                                ;
                                ;  Number of "extra" CL (communication line) units to be genned into
                                ;  system.  These CL units are not initially assigned to any line but
                                ;  may be used "take over" a time-sharing line to use it as a CL unit.
                                ;  The total number of CL units (those defined using CLDEF blocks plus
                                ;  the extra units) may not exceed 16. The first 8 CL units are
                                ;  named CL0 to CL7, the second 8 are named C10 through C17.
                                ;
000001                          CLXTRA  =       1.      ;Number of extra CL units.
                                ;
                                ;  Default output ring buffer size for I/O communication lines defined
                                ;  with the CLDEF macro and accessed as "CL" devices.
                                ;  The recommended value is ((3*baud_rate)/1000+2).
```

```
                                ;
000040                          CLORSZ  =       32.     ;Size of CL output ring buffers
                                ;
                                ;-----------------------------------------------------------------------
                                ;  Flags which can be used with the FLAGS macro within
                                ;  a line definition block to define line characteristics.
                                ;
100000                          $SCOPE  =       100000  ;ON==>CRT type terminal
040000                          $ECHO   =       40000   ;ON==>Echo characters to terminal
020000                          $TAPE   =       20000   ;ON==>"Paper-tape" mode (do x-on/x-off control, etc.)
010000                          $8BIT   =       10000   ;ON==>Support 8 bit (rather than 7 bit) characters.
004000                          $START  =       4000    ;ON==>Automatically start line during initialization
001000                          $TAB    =       1000    ;ON==>Do not simulate tabs (Terminal handles tab char)
000400                          $FORM   =       400     ;ON==>Do not simulate form-feeds (Terminal handles FF)
000200                          $AUTO   =       200     ;ON==>Do autobaud speed selection for line
000100                          $PAGE   =       100     ;ON==>Enable ctrl-S/ctrl-Q input processing
000040                          $LC     =       40      ;ON==>Enable lower-case input
000020                          $NOSUB  =       20      ;ON==>Disallow use of subprocesses
000010                          $DEFER  =       10      ;ON==>Do defered character echoing (recommended)
000004                          $QTSET  =       4       ;ON==>Set tt quiet (Don't list command files)
000002                          $SYSPS  =       2       ;ON==>Require system password before logon
000001                          $PHONE  =       1       ;ON==>Dial-up, modem connected line
                                ;
                                ;  Default line flags that will be used for each line that does
                                ;  not explicitly specify flags using a FLAGS macro.
                                ;
040150                          NRMFLG  =       $ECHO!$DEFER!$PAGE!$LC
                                ;
                                ;-----------------------------------------------------------------------
                                ;  Terminal type names that are legal to used with the TRMTYP macro
                                ;  within a line definition block to define the terminal type.
                                ;
                                ;  VT100 ==> DEC VT100
                                ;  VT200 ==> DEC VT200 with 7 bit control codes
                                ;  VT52  ==> DEC VT52
                                ;  LA36  ==> DEC LA36
                                ;  LA120 ==> DEC LA120
                                ;  HAZEL ==> Hazeltine brand terminals
                                ;  ADM3A ==> Lear Siegler ADM3A
                                ;  DIABLO==> Diablo brand terminals (with X-ON/X-OFF protocol)
                                ;  QUME  ==> Qume brand terminals (with X-ON/X-OFF protocol)
                                ;
```

```
                                ;-----------------------------------------------------------------
                                ;  Line definitions
                                ;
                                ;  The TBLDEF macro call requires four arguments:
                                ;    1.  The number of real (physical) time-sharing lines on machine.
                                ;    2.  The number of subprocess jobs.
                                ;    3.  The number of detached jobs.
                                ;    4.  The number of dedicated CL lines.
                                ;
003330                              TBLDEF  3.,2.,2.,0.     ;# Real, # Subprocess, # Detached, # CL lines

                                ;
                                ;  Define primary (real) time-sharing lines
                                ;


                                ;  #1 time-sharing line
006220                              LINDEF  60,177560,OPER          ;USE CONSOLE TERMINAL AS T/S TERM
006220                              NAME    <Console>
                                ;       CMDFIL  LINE1.TSX
                                ;       TRMTYP  VT100
006220                              FLAGS   NRMFLG!$START
006220                              LINEND

                                ;  #2 time-sharing line
006234                              LINDEF  310,176510
                                ;       CMDFIL  LINE2.TSX
                                ;       TRMTYP  LA120
006234                              FLAGS   NRMFLG
006234                              LINEND

                                ;  #3 time-sharing line
006250                              LINDEF  320,176520
                                ;       CMDFIL  LINE3.TSX
                                ;       TRMTYP  VT52
006250                              FLAGS   NRMFLG
006250                              LINEND
                                ;
                                ;;  The following section is an example of line definitions for a
                                ;;  DHV11 type multiplexer.
                                ;
                                ;       DHVDEF  370,160020              ;DHV11 MUX VECTOR & RSR ADDRESS
                                ;
                                ;;  Mux line # 0 - first line on DHV
                                ;       LINDEF  0
                                ;       CMDFIL  LINE2.TSX
                                ;       FLAGS   NRMFLG!$AUTO
                                ;       LINEND
                                ;
                                ;;  Mux line # 7 - last line on DHV
                                ;       LINDEF  7
                                ;       CMDFIL  LINE2.TSX
                                ;       FLAGS   NRMFLG!$AUTO
                                ;       LINEND
                                ;
                                ;       MUXEND                          ;END OF DHV11 MUX LINES USED
                                ;
                                ;;  The following section is an example of line definitions for a
```

```
                        ;;  DZV11 type multiplexer.
                        ;
                        ;       DZDEF   360,160010              ;DZV11 MUX VECTOR & RSR ADDRESS
                        ;
                        ;;  Mux line # 0 - first line on DZ
                        ;       LINDEF  0
                        ;       TRMTYP  VT100
                        ;       SPEED   S9600
                        ;       CMDFIL  LINE2.TSX
                        ;       LINEND
                        ;
                        ;;  Mux line # 3 - last line on DZ
                        ;       LINDEF  3
                        ;       TRMTYP  LA120
                        ;       SPEED   S1200
                        ;       CMDFIL  LINE2.TSX
                        ;       FLAGS   NRMFLG!$FORM
                        ;       LINEND
                        ;
                        ;       MUXEND          ;End of DZ11 lines
                        ;
                        ;  Use the "DETACH" macro here to declare any start-up command
                        ;  files and associated parameters (up to 80 characters) to be
                        ;  run as detached jobs:
                        ;
                        ;       DETACH  <SY:EXAMPL.TSX PARM1 PARM2> ;Detached job with parameters
                        ;       DETACH  <SY:DETACH.TSX>         ;Start-up detached job
                        ;       DETACH  <SY:WINPRT.TSX>         ;Start window-print detached job
                        ;
                        ;===============================================================
                        ;  END OF SECTION OF TSGEN TO BE ALTERED BY USER
                        ;===============================================================
```

```
                                        .LIST   MD
                                        .LIST   CND
                                        .ENDC
```

```
                                ;-----------------------------------------------
                                ;  Finish building tables
                                ;
                                ;  Make sure memory size parameters are reasonable.
                                ;
                                .IF     GT,HIMEM-64.
                                .ERROR  ;HIMEM may not exceed 64.
                                HIMEM   =       64.
                                        .ENDC
                                .IF     GT,DFLMEM-HIMEM
                                DFLMEM  =       HIMEM
                                        .ENDC
                                ;
                                ;  Make sure silo parameters are reasonable
                                ;  Actual silo limit is specified by MAXSLO
                                ;
                                        .IIF    GT,<NCSILO-255.> NCSILO = 255.
000016                          S = <NCSILO/2>-2
                                        .IIF    GT,<NCXOFF-S>    NCXOFF = S
000014                          NCXOFF  = NCXOFF&377    ;MUST BE BYTE SIZE
                                        .IIF    GT,<NCXON-S>     NCXON  = S
000004                          NCXON   = NCXON&377     ;MUST BE BYTE SIZE
                                ;
                                ;  Make sure last line definitions were properly terminated
                                ;
                                        .IF     NE BO
                                .ERROR  1 ; Missing LINEND for last line
                                        LINEND
                                        .ENDC
                                        .IF     NE CURMX
                                .ERROR  1 ; Missing MUXEND
                                        MUXEND
                                        .ENDC
                                ;
                                ;  Make sure the right # of lines were defined.
                                ;
                                        .IF     NE <NPLDF-NPL>
                                .ERROR  2; Wrong number of primary lines defined
                                        .ENDC
                                        .IF     NE <NCLDF-NIOL>
                                .ERROR  2; Wrong number of CL lines defined
                                        .ENDC
                                        .IF     GT <NDLDF-NDL>
                                .ERROR  2; Wrong number of detached jobs defined
                                        .ENDC
                                ;
                                ;  Define any additional detached job lines.
                                ;
000002                                  .REPT   <NPL+NDL-LN>
                                        DETACH
                                        .ENDR
                                        .IF     NE <LN-NPL-NDL>
                                .ERROR  2 ; Wrong number of lines defined
                                        .ENDC
                                        .IF     NE <NCLDF-NIOL>
                                .ERROR  2; Wrong number of CL lines defined
                                        .ENDC
```

```
                                ;
                                ;  Define tables for subprocesses (if any)
                                ;
                                        .IF     NE NSL
        000002                          .REPT   NSL
                                BO      =       1               ;BEGIN BLOCK
                                LN      =       LN+1
                                LX      =       LX+2
                                CLX     =       LX
                                        BUFSIZ  DINSPC,DOTSPC
                                        LINEND
                                        .ENDR
                                        .ENDC
006530                                  .CSECT  TSGEN
                                ;
                                ;  Define mux interrupt entry vectors
                                ;
                                ;  Input interrupt entry points
        000000                          .REPT   LSTMX/2
                                        INCB    MUXNUM
                                        .ENDR
006530  000167  000000G         INMXV:  JMP     ININT
                                ;  Output interrupt entry points (set up by TSINIT)
006534                          OTMXV:  .BLKW   3*<LSTMX/2>
                                ;
                                ;  Generate tables for Unibus map registers
                                ;
                                        .GLOBL  UMRBAS,UMREND,UMRWHD
006534  000000                  UMRWHD: .WORD   0
                                        .MACRO  UMRDEF  NUM
                                        .BYTE   CURUMR                  ;UM$UMR
                                        .BYTE   NUM                     ;UM$NMR
                                        .WORD   NUM*4096.               ;UM$WDS
                                        .WORD   0                       ;UM$IOQ
                                CURUMR  =       CURUMR+NUM
                                        .ENDM   UMRDEF
                                ;  Define UMR sets in order of size -- small to large.
        000005                  CURUMR  =       5               ;Map regs 0-4 always mapped by init code.
006536                          UMRBAS:
                                        .IF     EQ,<BUSTYP-UNIBUS>      ;Generate only for UNIBUS machines
                                        UMRDEF  1.
                                        UMRDEF  1.
                                        UMRDEF  4.
                                        UMRDEF  4.
                                        UMRDEF  8.
                                        UMRDEF  8.
                                        .ENDC
006536                          UMREND:
                                ;
                                ;  Check file and device caching parameters
                                ;
                                        .IF     LT,<MAXCSH-1>           ;MAKE SURE WE CACHE AT LEAST 1 DEVICE
                                MAXCSH  =       1
                                        .ENDC
                                        .IF     LT,<NMFCSH-4>           ;MINIMUM NUMBER OF CASHED FILES
                                NMFCSH  =       4
                                        .ENDC
```

```
                                ;
                                ;  Generate tables to keep track of allocated devices
                                ;
                                        .IF     LT <MAXALC-1>           ;Make sure we have at least 1 entry
                                MAXALC  =       1
                                        .ENDC
006536                          ALCTBL:                                 ;Base of device allocation table
        000005                          .REPT   MAXALC
                                        .WORD   0                       ;AD$DVU
                                        .BYTE   0                       ;AD$JOB
                                        .BYTE   0                       ;AD$FLG
                                        .ENDR
006562                          ALCEND:                                 ;End of device allocaton table
                                ;
                                ;  Check validity of PMSIZE
                                ;
                                        .IF     GT <PMSIZE-8192.> ;PMSIZE MAY NOT EXCEED 8192.
                                .ERROR 0;PMSIZE may not exceed 8192.
                                PMSIZE  =       8192.
                                        .ENDC
                                ;
                                ;  Check validity of MIOBSZ
                                ;
                                        .IF     GT <MIOBSZ-15.>
                                .ERROR 0;MIOBSZ may not exceed 15.
                                MIOBSZ  =       15.
                                        .ENDC
                                ;
                                ;  Check validity of CACHE parameter
                                ;
                                        .IF     GT <CACHE-4095.>
                                CACHE   =       4095.
                                        .ENDC
```

```
                                ;-----------------------------------------------
                                ;  Generate tables for shared file i/o control
                                ;
                                        .IF     NE,MAXSF        ;ANY SHARED FILES?
                                ;**
                                ;**  Assemble if there are shared files **
                                ;**
                                        .IFF
                                ;**
                                ;**  Assemble this code if there are no shared files **
                                ;**
                                MAXSFC  =       0
                                MXLBLK  =       0
                                NUMDC   =       0
                                ;
                                        .ENDC
                                ;
                                ;
                                ;
                                ;-------------------------------------------------------------------------
                                ;  Generate dummy shared run-time definitions
                                ;
000002                                  .REPT   NDRTDF
                                        RTDEF   <$$             >,R,0
                                        .ENDR
```

```
                                ;-------------------------------------------------------------------------------
                                ;  Generate tables for real-time support facility.
                                ;
                                ;  Generate the vector control blocks
                                ;
                                        .GLOBL  VCBBAS,VCBEND
006562                          VCBBAS:
                                        .REPT   RTVECT
000000                                  JSR     R2,@#RTINT
                                        .WORD   0,0,0           ;Must match size defined in TSDEFS
                                        .ENDR
006562                          VCBEND:
000050                          NUMIOQ  =       NUMIOQ+RTVECT   ;ADD I/O QUEUE ELEMENTS FOR INT COMPL ROUTINES
000017                          NUMSYQ  =       NUMSYQ+RTVECT
                                ;
                                        .IF     NE,RTVECT
                                ;**
                                ;**  Assemble this code if real-time support is wanted
                                ;**
                                        .GLOBL  RTINT
                                        .ENDC           ;End of real-time conditional
                                ;
                                ;-------------------------------------------------------------------------------
                                ;  Conditional code for different types of terminals.
                                ;
                                        .IF     NE,DHUSE
                                ;**
                                ;**  Assemble this code if DH11 support is needed
                                ;**
                                        .IFF
                                ;**
                                ;**  Assemble this code if DH11 support is not needed
                                ;**
                                        .GLOBL  TSDHIO,DHSTRT,DHSTOP,DHOINT
                                        .GLOBL  VHSTRT,VHOINT,DHTIMR,VHSTOP
                                        .GLOBL  DHXON,DHXOFF,VHXOFF,VHXON
000000                          TSDHIO  =       0
006562                          DHTIMR:
006562                          DHSTRT:
006562                          DHSTOP:
006562                          DHXON:
006562                          DHXOFF:
006562                          DHOINT:
006562                          VHSTRT:
006562                          VHSTOP:
006562                          VHXON:
006562                          VHXOFF:
006562                          VHOINT:
006562  000207                  RETURN
                                        .ENDC           ;End of DH11 conditional code
                                ;
                                ;-------------------------------------------------------------------------------
                                ;  Conditional code for CL units
                                ;
                                        .GLOBL  CL$OPT,CL$STA,CL$COL,CL$RQH,CL$WQH
                                        .GLOBL  CL$ORB,CL$ORA,CL$ORS,CL$ORG,CL$ORP,CL$ORE
                                        .GLOBL  CL$LEN,CL$LIN,CL$WID,CL$SKP,CL$LIX
```

```
                                        .GLOBL  CL$EPN,CL$EPS,CL$EPP,CL$XLN,CLSTS
                                ;
                                ;  Define CL device status word.  See the RT-11 .DSTATUS or .DRDEF macros
                                ;  for status bit definitions.  The device type code is the same as XL.
                                ;  (To make file names appear in SHOW QUEUE for spooled CL device,
                                ;   add the SPECL$ flag (10000).)
                                ;
006057                          CLSTS   = 4000+2000+57          ;HNDLR$+SPFUN$+XL$COD
                                ;
                                        .IF     NE,CLVRSN       ; Test to see if CLVRSN is defined as 0
                                        .IF     LE,<CLVRSN-14.> ; or greater than 14.
                                        .ERROR  0;Minimum CL version number is 15.  ;If not report an error
                                        CLVRSN  = 15.           ; and set to a reasonable number
                                        .ENDC                   ; End conditional LE,<CLVRSN-14.>
                                        .ENDC                   ; End conditional NE,CLVRSN
                                ;
                                ;  Define table that tells which line is associated with each CL unit.
                                ;  Note, this table is always generated.  Even if there are no CL lines
                                ;  genned in.
                                ;
006564  000000  000000  000000  CL$LIX:  .WORD   0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0 ;Line index for each CL unit
006572  000000  000000  000000
006600  000000  000000  000000
006606  000000  000000  000000
006614  000000  000000  000000
006622  000000
                                ;
                                ;  Macro to define CL tables which have one entry per CL unit
                                ;
                                        .MACRO  CLTABL NAME
                                NAME:
                                        .REPT   CLTOTL
                                        .WORD   0
                                        .ENDR
                                        .ENDM   CLTABL
                                ;
                                        .IF     NE,CLTOTL       ;Assemble if there are CL units
                                ;**
                                ;**  Assemble this code if there are CL units
                                ;**
006624                              CLTABL  CL$OPT          ;Option flags (CO$xxx)
006626                              CLTABL  CL$STA          ;Status flags (CM$xxx)
006630                              CLTABL  CL$COL          ;Current column position
006632                              CLTABL  CL$LEN          ;Number of lines per page
006634                              CLTABL  CL$LIN          ;Current line number
006636                              CLTABL  CL$XLN          ;Number of line CL unit is cross connected to
006640                              CLTABL  CL$SKP          ;Number of lines to skip at bottom of page
006642                              CLTABL  CL$WID          ;Maximum allowed line width
006644                              CLTABL  CL$RQH          ;Internal queue head for read requests
006646                              CLTABL  CL$WQH          ;Internal queue head for write requests
006650                              CLTABL  CL$ORB          ;Start of output ring buffer
006652                              CLTABL  CL$ORE          ;End of output ring buffer
006654                              CLTABL  CL$ORP          ;Pointer where next char goes in output ring
006656                              CLTABL  CL$ORG          ;Pointer to next char to get from output ring
006660                              CLTABL  CL$ORA          ;Allocated size of output ring buffer
006662                              CLTABL  CL$ORS          ;Free space in output ring buffer
006664                              CLTABL  CL$EPN          ;Number of Form-feeds for end page
```

```
                                        CLTABL  CL$EPS          ;Pointer to end-of-file string buffer
                                        CLTABL  CL$EPP          ;Pointer to next char within EOF string buffer
                                ;
                                        .IFF                    ;Assemble if no CL units
                                ;**
                                ;**  Assemble this code if there are no CL units
                                ;**
                                        .GLOBL  CLOTIR,CLSIZE,CLHEAD,CLLQE,CLCQE,CLINIR,CLABF
                                        .GLOBL  TSCLR
                                TSCLR   =       0
                                CLSIZE  =       0
                                CLHEAD  =       0
                                ;
                                CLCQE:
                                CLLQE:
                                CL$OPT:
                                CL$STA:
                                CL$COL:
                                CL$LEN:
                                CL$LIN:
                                CL$XLN:
                                CL$SKP:
                                CL$WID:
                                CL$RQH:
                                CL$WQH:
                                CL$ORB:
                                CL$ORE:
                                CL$ORP:
                                CL$ORQ:
                                CL$ORA:
                                CL$ORS:
                                CL$EPN:
                                CL$EPS:
                                CL$EPP:
                                        .WORD   0
                                ;
                                CLINIR:
                                CLOTIR:
                                CLABF:
                                        RETURN
                                ;
                                        .ENDC                   ;End conditional (NE,CLTOTL)
                                ;
                                ;-----------------------------------------------------------------------
                                ;  Define some misc data cells
                                ;
006672  000031                  NMFREQ: .WORD   NUMIOQ-NUMSYQ   ;# I/O queue elements available for user jobs
                                ;
                                ;  Invoke dummy SYSPS if user commented it out
                                ;
                                        .IIF    EQ,SYSPSS       SYSPS   <>
                                ;
                                ;  Make sure PHONE parameter is 0 or 1
                                ;
                                        .IIF    NE,PHONE        PHONE = 1
                                ;
                                ;  Close out some CSECTS.
```

```
    167                                 ;
    168 000034                                  .CSECT  RDBSEC
    169 000034                          RDBEND:
    170 000120                                  .CSECT  SASECT
    171                                         .EVEN
    172 000000                                  .CSECT  GENTOP
    173 000000  123456                  GENTOP: .WORD   123456          ;Flag word for top of tsgen
    174                                         .IF NE FULLST
    176                                         .LIST   BIN
    177                                         .ENDC ;NE FULLST
    178                                 ;
    179                                 ;  Define address of top of resident portion of TSX-Plus
    180                                 ;
    181 000000                                  .ASECT
    182          000050                 .       =       50
    183 000050   000000G                        .WORD   PROITP
    184          000014                 .       =       14
    185 000014   000000G                        .WORD   SCHED
    186 000016   000000G                        .WORD   DSKBUF
    187                                 ;
    188          000001                         .END
Errors detected:   0

*** Assembler statistics

Work  file  reads:  137
Work  file writes:  127
Size of work file:  21456 Words  ( 84 Pages)
Size of core pool:  17920 Words  ( 70 Pages)
Operating  system:  RT-11

Elapsed time:  00:01:39.44
DK:TSGEN,LP:TSGEN=DK:TSGEN.MAC/C/N:SYM
```

Cross reference table (CREF V05.04)

```
$8BIT    17-68#
$AUTO    17-72#
$DEFER   17-76#     17-84
$ECHO    17-66#     17-84
$FORM    17-71#
$INDDV    3-104#
$LC      17-74#     17-84
$MEMSZ    1-50       3-101#
$NOSUB   17-75#
$NOVLN    2-34#
$PAGE    17-73#     17-84
$PHONE   17-79#
$PRIV     2-33#
$QTSET   17-77#
$SCOPE   17-65#
$START   17-69#     18-21
$SYSPS   17-78#
$TAB     17-70#
$TAPE    17-67#
$TCFIG    3-103#
ABRTOV    1-126      4-93#
ADM3A     1-133
AHEND     1-89      16-551#
ALCEND    1-86      20-127#
ALCTBL    1-86      20-121#
AUTHAN    1-89      16-542#
BASMAP    1-110      4-325#
BLKEY     1-98       3-61#
BLKMV     1-134      3-224
BMJMP     3-222      3-224#
BO        8-50#     18-17     18-17#    18-22     18-22#    18-25     18-25#    18-29     18-29#    18-32     18-32#    18-36
         18-36#     20-26     20-51     20-51     20-69     20-69     20-69#    20-69#    20-69#    20-69#
BOTCSR    1-88       4-103#
BOTDEV    1-88       4-101#
BOTUNI    1-88       4-102#
BUSTYP    4-174     16-90#    20-97
C1DEVX    1-105      4-100#
CA$BLK    1-120      4-122#
CA$DVU    1-120      4-123#
CA$HBL    1-120      4-128#
CA$HFL    1-120      4-127#
CA$HSH    1-120      4-129#
CA$UBL    1-120      4-126#
CA$UFL    1-120      4-125#
CA$WCT    1-120      4-124#
CACHE     4-37       4-38      16-363#   20-145
CASCBR    1-122      4-137#
CASCUP    1-122      4-140#
CASTBR    1-122      4-136#
CASTBW    1-122      4-139#
CASTRO    1-122      4-135#
CASTWO    1-122      4-138#
CCBHD     1-121      4-134#
CCLSAV    1-103      4-61#
CCXCTL    4-190     16-461#
CCXTRM    4-189     16-454#
```

Cross reference table (CREF V05.04)

```
CDX$DH    1-78      2-76#
CDX$DL    1-78      2-74#     2-92
CDX$DZ    1-78      2-75#
CDX$PC    1-78      2-79#
CDX$PI    1-77      2-78#
CDX$PP    1-78      2-80#
CDX$QP    1-79      2-81#
CDX$VH    1-78      2-77#
CFABLV    1-50      3-91#
CFCHAN    1-69      2-83#
CFSTS     1-50      3-87#     3-210
CHKEY     1-98      3-62#
CL$COL   22-55     22-95#
CL$EPN   22-58     22-109#
CL$EPP   22-58     22-111#
CL$EPS   22-58     22-110#
CL$LEN   22-57     22-96#
CL$LIN   22-57     22-97#
CL$LIX   22-57     22-78#
CL$OPT   22-55     22-93#
CL$ORA   22-56     22-107#
CL$ORB   22-56     22-103#
CL$ORE   22-56     22-104#
CL$ORG   22-56     22-106#
CL$ORP   22-56     22-105#
CL$ORS   22-56     22-108#
CL$RQH   22-55     22-101#
CL$SKP   22-57     22-99#
CL$STA   22-55     22-94#
CL$WID   22-57     22-100#
CL$WQH   22-55     22-102#
CL$XLN   22-58     22-98#
CLDEVX    1-105     4-99#
CLEOFS    1-55      2-31#
CLKVEC    1-126     2-27#
CLORSZ    1-61     17-59#
CLSCDB    1-125     4-164#
CLSTS    22-58     22-65#
CLTOTL    1-64     18-10#    22-89     22-93     22-94     22-95     22-96     22-97     22-98     22-99     22-100    22-101
         22-102    22-103    22-104    22-105    22-106    22-107    22-108    22-109    22-110    22-111
CLVERS    1-61      4-208#
CLVRSN    4-208    16-468#   22-67
CLX      18-17     18-17     18-17     18-17     18-17     18-17     18-17     18-17#    18-18     18-22     18-22     18-22
         18-22     18-22     18-22     18-22     18-22     18-22     18-22     18-25     18-25     18-25     18-25     18-25
         18-25     18-25#    18-29     18-29     18-29     18-29     18-29     18-29     18-29     18-29     18-29     18-29
         18-29     18-32     18-32     18-32     18-32     18-32     18-32     18-32#    18-36     18-36     18-36     18-36
         18-36     18-36     18-36     18-36     18-36     18-36     18-36     20-51     20-51     20-51#    20-51#    20-69
         20-69     20-69     20-69     20-69     20-69     20-69#    20-69#
CLXTRA    1-64     17-53#    18-10     18-10
CMFDON   18-17#    18-22     18-22#    18-25#    18-29     18-29#    18-32#    18-36     18-36#    20-69     20-69
CMFSEC    1-72      9-81#
CMFTOP    1-72     18-22#    18-29#    18-36#
CONFG2    1-105     3-88#
CONFIG    1-99      3-74#
CORTIM    1-109     4-11      16-319#
CORUSR    1-93      3-81#      3-206
```

Cross reference table (CREF V05.04)

```
CP$RT     2-66       2-69#
CP$STD    2-66       2-68#
CP$SYN    2-66       2-70#
CSHALC    1-62       4-38#
CSHBAS    1-123      4-69#
CSHBFP    1-121      4-130#
CSHCLN    1-123      4-145#
CSHDEV    1-85       4-95#
CSHDVN    1-85       4-96#
CSHFHD    1-121      4-133#
CSHFIN    1-123      4-146#
CSHHD     1-85       4-77#
CSHINI    1-123      4-143#
CSHIO     1-123      4-144#
CSHLRU    1-121      4-131#
CSHMRU    1-121      4-132#
CSHSIZ    1-116      4-337#
CSHVEC    1-123      4-142#
CTRLTT    1-63       4-56#      18-17
CURCDX    2-92#      18-17      18-25      18-32
CURMX     2-90#      18-17      18-25      18-32      20-30
CURMXL    18-17#     18-22      18-25#     18-29      18-32#     18-36
CURUMR    20-95#
CVTPHY    1-134      3-219
DBGFLG    4-181      16-245#
DCAGE     1-81       2-28#
DCCRD     1-67       4-232#
DCCWR     1-67       4-234#
DCRD1     1-126      4-165#
DCRD2     1-126      4-166#
DCTOTU    1-67       4-230#
DCTRD     1-67       4-231#
DCTWR     1-67       4-233#
DEFBAS    4-107#
DETCBS    1-52       2-8#       20-51      20-51
DEVSIZ    1-101      3-165#
DEVXMR    4-33       16-574#
DFJMEM    1-111      4-332#
DFLAGS    18-17#     18-21#     18-22      18-25#     18-28#     18-29      18-32#     18-35#     18-36
DFLG      1-98       3-64#
DFLMEM    4-29       16-35#     20-10
DHBFSZ    1-75       2-14#
DHOINT    22-34      22-43#
DHSTOP    22-34      22-40#
DHSTRT    22-34      22-39#
DHTIMR    22-35      22-38#
DHUSE     2-93#      22-26
DHXOFF    22-36      22-42#
DHXON     22-36      22-41#
DIABLO    1-134
DINSPC    4-45       17-10#     18-17      18-25      18-32      20-69      20-69
DIS       18-17#     18-22      18-25#     18-29      18-32#     18-36      20-69      20-69      20-69#     20-69#
DLINT     1-130      18-10      18-10      18-10
DM$CSR    4-264      4-264#     4-282
DM$LSR    4-265      4-265#     4-281
DM$VEC    4-266      4-266#
```

Cross reference table (CREF V05.04)

```
DMA       5-19#
DMPKTP    4-172     16-168#
DMPTCR    4-115     16-161#
DOCOPN    1-124      4-154#
DOCULK    1-125      4-159#
DOOPAP    1-124      4-153#
DORLK     1-124      4-157#
DOS      18-17#     18-22     18-25#    18-29     18-32#    18-36     20-69     20-69     20-69#    20-69#
DOSFCK    1-125      4-161#
DOTLK     1-125      4-158#
DOTSPC    4-46      17-11#     18-17     18-25     18-32     20-69     20-69
DOULK1    1-125      4-160#
DSKBUF    1-136      4-41      22-186
DTYPE     1-48      16-542#
DVFLAG    1-114      3-174#
DVFLG    16-543     16-543#    16-544    16-544#   16-545    16-545    16-545#   16-545#   16-546    16-546    16-546#   16-546#
         16-547     16-547     16-547#   16-547#   16-548    16-548#   16-549    16-549#   16-550    16-550#   16-551    16-551
         16-551     16-551     16-551#   16-551#   16-551#   16-551#
DVNUM     5-38#     16-543     16-543    16-543#   16-544    16-544    16-544#   16-545    16-545    16-545#   16-546    16-546
        16-546#     16-547     16-547    16-547#   16-548    16-548    16-548#   16-549    16-549    16-549#   16-550    16-550
        16-550#     16-551     16-551    16-551    16-551    16-551    16-551    16-551    16-551    16-551    16-551#   16-551#
        16-551#     16-551#
DVSTAT    1-100      3-156#
DWTYPE    1-79       3-109#
EDIT      4-349#
EDITOR    4-191     16-490#
EMTRTN    3-93#
EVEN      4-381#
EVNBUF    5-21#
EXTCHN    1-113      3-182#    3-193     3-204
EXTP1     1-134      3-223
FASTIN    1-53       2-26#
FIRST     4-356#
FNDHRB    1-130      3-220
FORKQ     1-135      3-17
FP$CDI    2-44       2-50#
FP$CDO    2-44       2-51#
FP$CK1    2-44       2-56#
FP$CKT    2-43       2-49#
FP$DEF    2-43       2-52#
FP$FLG    2-44       2-46#     2-47      2-48      2-49      2-50      2-51      2-52      2-53      2-54      2-55      2-56
          2-58       2-59
FP$IOA    2-43       2-54#
FP$IOF    2-43       2-53#
FP$IOS    2-43       2-58#
FP$MAX    2-44       2-47#
FP$MOV    2-43       2-59#
FP$PIO    2-44       2-55#
FP$RT     2-43       2-48#
FREFRK    1-88      4-289#
FREIOQ    1-102     4-237#
FREPGS    1-110     4-329#
FREUMR    1-47       3-20
FRKADR    3-94#
FRKDLY    1-94       4-55#
FRKGEN    1-56       2-9#      4-292
```

Cross reference table (CREF V05.04)

```
FRKGET   1-135     3-16
FRKINI   1-56      4-290#
FRKLST   4-289     4-291#
FSTDL    1-92     18-10#
FSTIOL   1-49     18-10     18-10#
FSTSL    1-63     18-10#
FULLST   1-39#     1-41     15-1      19-1      19-2      22-174
GENTOP   1-109    22-173#
GETRTQ   1-135     3-11
GETUMR   1-47      3-19
HANBUF   5-27#
HANDSK   1-46      3-159#
HANENT   1-100     3-9       3-152#
HANIOC   1-101     3-171#
HANPAR   1-119     3-10      3-168#
HANRCB   1-46      4-118#
HANRCO   1-46      3-231#
HANSIZ   1-100     3-162#
HANXMR   1-134     3-221
HAZEL    1-134
HIMAP    1-110     4-327#
HIMEM    3-68      4-28     16-30#    20-6      20-10
HIPRCT   4-12     16-310#
HNMEPT   3-105     3-229#
HNSPDO   5-28#
HSUFFX   3-97#
I        18-10     18-10     18-10     18-10     18-10     18-10     18-10     18-10     18-10     18-10     18-10     18-10
         18-10     18-10     18-10     18-10     18-10     18-10     18-10#    18-10#    18-10#    18-10#    18-10#    18-10#
         18-10#    18-10#    18-10#    18-10#    18-10#    18-10#
ILSW2    18-10     18-10#    18-22     18-29     18-36
INDDBL    1-104     4-63#
INDDBS    1-104     4-64#
INDDEV    3-115#
INDFIL    1-91     16-25#
INDOFF    3-104     3-114#
INDSAV    1-103     4-62#
INDTSV    1-104     3-189#    3-211
INIABT    4-175    16-120#
ININT     1-130    18-10     20-79
INMXV     1-59     20-79#
INRECV    1-57     18-10#
INSTA     3-100#    3-209
INSTBL    1-51      4-91#
INSTBN    1-51      4-92#
INTEN     1-132     3-38
INTIOC    4-13     16-303#
INTSSZ    1-102     2-20#
INVEC     1-56     18-10     18-10#    18-17     18-25     18-32
IOABFL    1-63      4-106#
IOABT     4-106    16-178#
IOFIN     1-132     3-69
IOHANQ    1-47      3-18
IOHLTM    1-54      2-30#
IOLFLG    8-52#    18-17     18-17     18-22     18-22     18-22     18-22     18-22#    18-25     18-25     18-29     18-29
         18-29     18-29     18-29#    18-32     18-32     18-36     18-36     18-36     18-36     18-36#    20-69     20-69
         20-69     20-69     20-69#    20-69#
```

Cross reference table (CREF V05.04)

```
IOLN      8-53#
IOQSIZ    1-136     3-22
IOSTRT    1-135     3-14
ITRMTP   18-10     18-10#
JCXPGS    1-111     4-330#
K52       4-352#
KED       4-351#   16-490
KEYMAX    4-35     16-230#
KMNHI     1-53      4-60#
KMNTOP    1-53      4-59#
KUSECK    1-119     2-29#
L        16-551    16-551    16-551#
LA120     1-133
LA36      1-133
LABTIM   18-10     18-10#
LACTIV   18-10     18-10#
LAFSIZ   18-10     18-10#
LAST      4-358#
LBASE    18-10     18-10#
LBRKCH   18-10     18-10#
LBRKCQ   18-10     18-10#
LBSPRI   18-10     18-10     18-10#
LCDTIM   18-10     18-10#
LCDTYP   18-10     18-10#    18-17     18-25     18-32
LCLUNT   18-10     18-10#
LCMPL    18-10     18-10#
LCMQHD   18-10     18-10#
LCOL     18-10     18-10#
LCONTM   18-10     18-10#
LCPUHI   18-10     18-10#
LCPULO   18-10     18-10#
LCXPAR   18-10     18-10#
LCXTBL   18-10     18-10#    18-22     18-29     18-36
LDDEVX    1-105     4-98#
LDHB1B   18-10     18-10#
LDHB1P   18-10     18-10#
LDHB2B   18-10     18-10#
LDHB2R   18-10     18-10#
LDHB2S   18-10     18-10#
LDSYS     4-179   16-212#
LDVERS    1-61     4-209#
LEMTPC   18-10     18-10#
LESCHR   18-10     18-10#
LESRTN   18-10     18-10#
LF       18-22     18-22#    18-22#    18-29     18-29#    18-29#    18-36     18-36#    18-36#    20-69     20-69     20-69#
         20-69#
LFWLIM   18-10     18-10#
LHIPCT   18-10     18-10#
LHIRBA   18-10     18-10#    18-22     18-29     18-36
LHIRBB   18-10     18-10#
LHIRBC   18-10     18-10#    18-22     18-29     18-36
LHIRBE   18-10     18-10#
LHIRBG   18-10     18-10#
LHIRBP   18-10     18-10#
LHIRBS   18-10     18-10#
LINBUF   18-10     18-10#
```

```
LINCNT  18-10    18-10#
LINCUR  18-10    18-10#
LINEND  18-10    18-10#
LINIR   18-10    18-10#
LINNUM   1-93     1-130    4-196#   18-10*   18-10*   18-10*
LINNXT  18-10    18-10#
LINPNT  18-10    18-10#
LINSIZ  18-10    18-10#   18-22    18-29    18-36    20-69    20-69
LINSPC  18-10    18-10#
LIOCNT  18-10    18-10#
LIOHLD  18-10    18-10#
LITIME  18-10    18-10#
LJSW    18-10    18-10#
LMEMIN  18-10    18-10#
LMINQ   18-10    18-10#
LMONHD  18-10    18-10#
LMSGBF  18-10    18-10#
LMXLN    1-59    18-10#
LMXNUM  18-10    18-10#
LMXPRM   1-59    18-10    18-10#   18-22    18-29    18-36
LN       8-49#   18-17    18-17    18-17#   18-22    18-25    18-25    18-25#   18-29    18-32    18-32    18-32#
        18-36    20-49    20-51    20-51    20-51    20-51    20-51#   20-51#   20-52    20-69    20-69    20-69
        20-69    20-69#   20-69#
LNAME   18-10    18-10#   18-18    18-29    18-36
LNBLKS  18-10    18-10#
LNMAP    1-60    18-10#
LNMTOP   1-48    18-18#   18-29#   18-36#
LNPRIM   1-60    18-10#
LNSBLK  18-10    18-10#
LNSPAC  18-10    18-10#
LOFFTM  18-10    18-10#
LOGCHN   1-68     2-84#
LOKBAS   1-82     4-68#
LOKCSH   1-82     4-72#
LOKINI   1-124    4-152#
LOKMEM   1-82     4-71#
LOKVEC   1-124    1-126    4-151#
LOMAP    1-110    4-326#
LOTBUF  18-10    18-10#
LOTEND  18-10    18-10#
LOTNXT  18-10    18-10#
LOTPNT  18-10    18-10#
LOTSIZ  18-10    18-10#   18-22    18-29    18-36    20-69    20-69
LOTSPC  18-10    18-10#
LOUTIR  18-10    18-10#
LPARBS  18-10    18-10#
LPARNT  18-10    18-10#
LPRG1   18-10    18-10#
LPRG2   18-10    18-10#
LPRI     1-60    18-10#
LPROG   18-10    18-10#
LPROJ   18-10    18-10#
LQLINK  18-10    18-10#
LQUAN   18-10    18-10#
LRBFIL  18-10    18-10#
LRDTIM  18-10    18-10#
```

Cross reference table (CREF V05.04)

```
LRTCHR  18-10     18-10#
LSCCA   18-10     18-10#
LSECPT  18-10     18-10#    18-22     18-29     18-36
LSLEPH  18-10     18-10#
LSLEPL  18-10     18-10#
LSNDCH  18-10     18-10#
LSPACT  18-10     18-10#    18-22     18-29     18-36     20-69     20-69
LSPND   18-10     18-10#
LSTACT  18-10     18-10#
LSTATE  18-10     18-10#
LSTDL    1-92     18-10     18-10#
LSTHL    1-49     18-10#
LSTIOL   1-57     18-10#
LSTLIN   1-58     18-10#
LSTMX    1-58      2-89#    20-76     20-81
LSTPL    1-57     18-10     18-10     18-10#
LSTPRM   9-137#   18-22     18-22     18-29     18-29     18-36     18-36
LSTSL    1-63     18-10     18-10#
LSUCF   18-10     18-10#    18-22     18-29     18-36     20-51     20-51
LSW     18-10     18-10#
LSW10   18-10     18-10#
LSW11   18-10     18-10#
LSW2    18-10     18-10#
LSW2S   18-10     18-10#
LSW3    18-10     18-10#
LSW4    18-10     18-10#
LSW5    18-10     18-10#
LSW6    18-10     18-10#
LSW7    18-10     18-10#
LSW8    18-10     18-10#
LSW9    18-10     18-10#
LSWPBK  18-10     18-10#
LTRMTP  18-10     18-10#
LTSCMD  18-10     18-10#
LTTCR   18-10     18-10#
LTTPAR  18-10     18-10#
LUNAME   1-57     18-10#
LWINDO  18-10     18-10#
LX       8-51#    18-17     18-17     18-17#    18-25     18-25     18-25#    18-32     18-32     18-32#    20-51     20-51
        20-51     20-51     20-51#    20-51#    20-69     20-69     20-69     20-69     20-69#    20-69#
LXCL    18-10     18-10#
LXX     18-10     18-10     18-10     18-10     18-10     18-10     18-10#    18-10#    18-10#    18-10#
MAPH     5-25#
MAPIO    5-20#    16-546    16-547
MAPPAR   1-84      4-328#
MAPSIZ   1-113     4-324#
MAPUSR   1-92      4-195#
MAXALC   1-86     16-387#   20-118    20-122
MAXBLK   1-99      3-76#
MAXCSH   4-94     16-377#   20-109
MAXDEV   1-101     2-5#      3-149     3-152     3-156     3-159     3-162     3-165     3-168     3-171     3-174     16-543
        16-544    16-545    16-546    16-547    16-548    16-549    16-550    16-551    16-551    16-551    16-551    16-551
MAXFIL   3-76      4-19     16-354#
MAXGVL   1-105     3-177#
MAXMC    4-24     16-640#
MAXMON   4-21     16-391#
```

```
MAXMRB    4-27      16-652#
MAXMSG    4-26      16-648#
MAXMUX    2-13#      4-251     4-252     4-253     4-254     4-255     4-256     4-257     4-258     4-259     4-260     4-261
          4-262      4-263     4-264     4-265     4-266
MAXSEC    1-64      16-349#   18-22     18-22     18-29     18-29     18-36     18-36
MAXSF     4-14      16-610#   21-4
MAXSFC    4-15       4-74     16-617#
MAXWIN    4-34      16-239#
MEM256    1-87       4-205#
MEMPTR    1-50       3-105#
MEMSIZ    4-324     16-100#
MFPMOV    3-126#     3-213
MFPS      3-85       3-125#
MH$BAR    1-75       4-275#
MH$BCR    1-75       4-274#    4-285
MH$BRK    4-261      4-261#
MH$CAR    1-74       4-273#    4-283
MH$LPR    4-262      4-262#    4-280
MH$PBR    4-263      4-263#
MH$RCR    1-74       4-272#    4-279
MH$SCR    1-74       4-271#    4-278
MH$SSR    1-75       4-276#    4-284
MHNSIZ    1-119      4-335#
MIDDLE    4-357#    16-205
MINTIM    1-94       4-57#
MIOBHD    1-52       4-79#
MIOBSZ    4-193     16-566#   20-138
MIODBG    1-84       2-22#
MIOFLG    1-88       4-206#
MIONBF    4-192     16-561#
MIONWB    1-91       2-21#
MIOSYQ    1-116      4-81#
MIOWHD    1-116      4-80#
MODDAT    1-70       4-116#
MODTIM    1-72       4-117#
MONAME    3-96#
MONFQH    1-61       4-78#
MONVEC    1-103      3-38#     3-95      3-105     3-111     3-114     3-177
MPARFL    1-117      2-23#
MSCHRS    4-25      16-644#
MSGBAS    1-80       4-66#
MTPS      3-84       3-141#
MUXNUM    1-93       1-131     4-197#
MVSIZ     1-69       3-197#    3-198
MVWDS     1-113      3-198#
MXBRK     1-74       4-269#
MXCAR     4-258      4-258#    4-269     4-276
MXCSR     4-252      4-252#    4-271
MXDTR     4-255      4-255#    4-274
MXJADR    1-108      4-333#
MXJMEM    1-111      4-331#
MXLBLK    4-17      16-623#
MXLNT     4-260      4-260#
MXLPR     4-253      4-253#    4-267
MXRBUF    1-74       4-267#    4-272
MXRING    1-74       4-268#
```

```
MXSBRK    4-257     4-257#
MXSPAC    1-73      16-473#   18-22     18-29     18-36     20-69     20-69
MXTBUF    4-256     4-256#    4-268     4-275
MXTCR     4-254     4-254#    4-273
MXTTCT    1-46      16-481#   18-22     18-22     18-29     18-29     18-36     18-36
MXTYPE    4-251     4-251#
MXVEC     4-259     4-259#
NAMDON    18-17#    18-18#    18-22     18-25#    18-29     18-29#    18-32#    18-36     18-36#    20-69     20-69
NAMPTR    18-18     18-18#    18-22     18-22#    18-29     18-29     18-29#    18-29#    18-36     18-36     18-36#    18-36#
NAMSEC    1-48      9-27#
NCLDF     8-55#     20-40     20-55
NCSILO    1-62      4-42      17-32#    20-17     20-18
NCXOFF    1-62      4-43      17-38#    20-19     20-20     20-20#
NCXON     1-62      4-44      17-44#    20-21     20-22     20-22#
NDL       1-92      18-10     18-10     18-10     18-10     18-10     18-10     18-10     18-10     18-10     18-10     18-10
          18-10     18-10     18-10#    20-43     20-49     20-51     20-51     20-51     20-51     20-52
NDLDF     8-56#     20-43     20-51     20-51     20-51#    20-51#
NDRTDF    2-32#     21-23
NDVRCB    1-59      4-33#
NESB      1-53      16-598#
NFRESB    1-55      4-88#
NGR       4-32      16-79#
NIOL      1-49      18-10     18-10     18-10     18-10     18-10     18-10     18-10     18-10     18-10     18-10     18-10
          18-10     18-10     18-10     18-10     18-10     18-10#    20-40     20-55
NLCHN     1-50      2-88#     3-183
NLIN2     1-49      18-10     18-10#
NLINES    1-50      18-10     18-10     18-10     18-10     18-10     18-10     18-10     18-10     18-10     18-10     18-10
          18-10     18-10     18-10     18-10     18-10     18-10     18-10     18-10     18-10     18-10     18-10     18-10
          18-10     18-10     18-10     18-10     18-10     18-10     18-10     18-10     18-10     18-10     18-10     18-10
          18-10     18-10     18-10     18-10     18-10     18-10     18-10     18-10     18-10     18-10     18-10     18-10
          18-10     18-10     18-10     18-10     18-10     18-10     18-10     18-10     18-10     18-10     18-10     18-10
          18-10     18-10     18-10     18-10     18-10     18-10     18-10     18-10     18-10     18-10     18-10     18-10
          18-10     18-10     18-10     18-10     18-10     18-10#
NMFCSH    4-20      16-382#   20-112
NMFREQ    1-102     22-156#
NMSNMB    1-109     2-10#     4-76
NMSYMB    2-11#     4-76
NMUMB     1-109     4-76#
NOCACH    5-22#
NODMA     5-31#     5-32
NOMAPH    5-26#
NOMOUN    5-23#
NONDMA    5-32#
NONE      4-383#    9-137
NOSET     5-29#     16-545
NPL       1-49      18-10     18-10     18-10     18-10     18-10     18-10     18-10     18-10     18-10     18-10     18-10
          18-10     18-10     18-10     18-10     18-10     18-10     18-10     18-10     18-10     18-10     18-10     18-10#
          18-17     18-22     18-25     18-29     18-32     18-36     20-37     20-49     20-51     20-51     20-52     20-69
          20-69
NPLDF     8-54#     18-17     18-17#    18-25     18-25#    18-32     18-32#    20-37
NRMFLG    4-39      17-84#    18-17     18-21     18-25     18-28     18-32     18-35
NSCP      1-65      2-19#
NSL       1-49      18-10     18-10     18-10     18-10     18-10     18-10     18-10     18-10     18-10     18-10     18-10
          18-10     18-10     18-10#    20-61     20-62
NSPLBL    1-60      4-87#
NSPLDV    1-55      4-207#
```

```
NSPLFL    1-60      4-86#
NUCHN     1-69      2-12#     2-83      2-84      2-85      2-86      2-87      2-88
NUIP      4-36     16-249#
NUMCCB    1-107     2-18#
NUMCDB    1-81      4-74#
NUMDC     4-16      4-73     16-630#
NUMDCD    1-81      4-73#
NUMDEV    1-101     3-8       4-236#
NUMFRK    1-57      2-6#
NUMIOQ    1-102     2-16#    22-13     22-13#    22-156
NUMON     1-93      4-198#
NUMRDB    1-84      2-91#    21-25     21-25     21-25#    21-25#
NUMSYQ    1-102     2-17#    22-14     22-14#    22-156
NXIVMH    1-57      2-7#
ODD       4-382#
OFFTIM    4-23     16-419#
OTMXV     1-59     20-81#
OTRASZ    1-60     17-17#
OTRECV    1-57     18-10#
P1EXT     3-106#
P1XPTR    3-106     3-223#
PHONE     4-194    16-431#   22-164
PHYMEM    1-50      3-148#
PIDPTR    1-117     4-112#
PMSIZE    1-73     16-675#   20-131
PNAME     1-100     3-95      3-149#
PNPTR     3-95#
PRIDEF    4-184    16-338#
PRIHI     4-183    16-334#
PRILOW    4-182    16-333#
PRIVIR    4-185    16-345#
PROBRK    1-118     2-24#
PROFLG    1-117     4-201#
PROITP    1-130    22-183
PROODC    1-118     2-25#
PROSLT    1-100     4-113#
PSW       1-132     3-126
PVON      1-93      4-199#
PVSPBL    1-59     16-598    16-598    16-598#
PWCH      4-188    16-448#
QBUS      1-114     4-300#   16-90
QCOMP     1-98      3-69#
QCOMPL    1-135     3-15
QFREE     1-135     3-12
QIO       1-135     3-13
QUAN0     4-4      16-258#
QUAN1     1-58      4-5      16-267#
QUAN1A    1-68      4-6      16-273#
QUAN1B    1-68      4-7      16-277#
QUAN1C    4-8      16-282#
QUAN2     1-58      4-9      16-290#
QUAN3     4-10     16-296#
QUME      1-134
R$CFST    1-58      3-210#
R$CH17    1-112     3-203#
R$CHN     1-112     3-202#
```

Cross reference table (CREF V05.04)

```
R$DATE    1-112     3-205#
R$INST    1-58      3-209#
R$INTC    1-104     3-211#
R$JOB     1-112     3-206#
R$MFMV    1-92      3-213#
R$SWPC    1-119     3-212#
R$TTOP    1-58      3-208#
R$UBAS    1-112     3-207#
R$XCHN    1-113     3-204#
RBR       1-56      18-10     18-10     18-10#    18-17     18-25     18-32
RDB       1-84      4-343#
RDBEND    1-84      22-169#
RDBSEC    1-48      4-342#
REQALC    5-24#
RF$WRT    1-109
RMNBAS    1-131     3-103     3-104
RPRCSR    1-79      3-107#
RPRVEC    1-79      3-108#
RSFBLK    1-90      16-17#
RSR       1-56      18-10     18-10#    18-17     18-25     18-32
RTVECT    1-94      16-666#   22-8      22-13     22-14     22-16
RUNCHN    1-76      2-86#
S         18-17     18-17     18-17     18-17#    18-17#    18-17#    18-18     18-18#    18-22     18-22     18-22     18-22
          18-22     18-22     18-22     18-22     18-22     18-22     18-22     18-22#    18-22#    18-22#    18-22#    18-22#
          18-22#    18-22#    18-22#    18-22#    18-25     18-25     18-25     18-25#    18-25#    18-25#    18-29     18-29
          18-29     18-29     18-29     18-29     18-29     18-29     18-29     18-29     18-29     18-29     18-29#    18-29#
          18-29#    18-29#    18-29#    18-29#    18-29#    18-29#    18-29#    18-29#    18-32     18-32     18-32     18-32#
          18-32#    18-32#    18-36     18-36     18-36     18-36     18-36     18-36     18-36     18-36     18-36     18-36
          18-36     18-36     18-36#    18-36#    18-36#    18-36#    18-36#    18-36#    18-36#    18-36#    18-36#    18-36#
          20-18#    20-19     20-21     20-51     20-51     20-51     20-51     20-51     20-51     20-51#    20-51#    20-69
          20-69     20-69     20-69     20-69     20-69     20-69#    20-69#    20-69#    20-69#    20-69#    20-69#
S110      4-364#
S1200     4-369#
S134.5    4-365#
S150      4-366#
S1800     4-370#
S19200    4-377#
S2000     4-371#
S2400     4-372#
S300      4-367#
S3600     4-373#
S4800     4-374#
S50       4-362#
S600      4-368#
S7200     4-375#
S75       4-363#
S9600     4-376#    9-137
SASECT    1-48      10-6#
SCHED     1-136     4-40      22-185
SCPFHD    1-51      4-97#
SD$HLD    1-131
SDBULS    1-91      16-598#
SDCB      1-54      16-598#
SDCBND    1-54      16-598#
SDCBSZ    1-91      16-598    16-598#
SEGBLK    4-31      16-75#
```

Cross reference table (CREF V05.04)

```
SEGCHN    1-90       4-58#
SETERR    1-131
SFCB      1-68       4-83#
SFCBFH    1-53       4-85#
SFCBND    1-68       4-84#
SFCLS     1-125      4-162#
SFRSST    1-124      4-156#
SFSVST    1-124      4-155#
SFWRIT    1-125      4-163#
SHRRCB    1-51       4-89#
SHRRCN    1-51       4-90#
SILSIZ   18-17#     18-22      18-25#     18-29      18-32#     18-36
SILXOF   18-17#     18-22      18-25#     18-29      18-32#     18-36
SILXON   18-17#     18-22      18-25#     18-29      18-32#     18-36
SIZMEM    4-324      4-324      4-324      4-324      4-324#
SLEDIT    4-180     16-221#
SMONHD    1-48       4-82#
SMRSIZ    1-116      4-334#
SNBUX     1-89      16-598     16-598     16-598     16-598#
SNDBX     1-87       4-87      16-598#
SNMSHD    1-108      4-75#
SPLANM    1-91      16-598#
SPLBHD    1-53      16-598#
SPLBLK    1-55      16-13#
SPLCHN    1-101     16-598#
SPLDEV    1-54      16-598     16-598#
SPLDVN    1-54      16-598     16-598#
SPLNB     1-52      16-598     16-598     16-598#
SPLND     1-52      16-598     16-598     16-598#
SPLNF     1-52       4-86      16-598     16-598#
SPOLID    1-118      4-104#
SPSTAT    1-61       3-98#
SPUSR     1-99       3-70#
SR3FLG    1-115      4-204#
SRERR    18-17      18-17#     18-25      18-25#     18-32      18-32#
SRTSIZ    1-116      4-336#
STPFLG    1-94       4-202#
SWAPFL    4-173     16-51#     18-10
SWDBLK    1-65      16-9#
SWPCHN    1-102      3-193#     3-212
SWPSLT    4-30      16-65#     18-10      18-10      18-10#
SYCH0     1-95       3-43#
SYCH1     1-95       3-44#
SYCH10    1-96       3-51#
SYCH11    1-96       3-52#
SYCH12    1-96       3-53#
SYCH13    1-96       3-54#
SYCH14    1-97       3-55#
SYCH15    1-97       3-56#
SYCH16    1-97       3-57#
SYCH17    1-97       3-58#      3-203
SYCH2     1-95       3-45#
SYCH20    1-97       3-59#
SYCH3     1-95       3-46#
SYCH4     1-95       3-47#
SYCH5     1-95       3-48#
```

```
SYCH6     1-96       3-49#
SYCH7     1-96       3-50#
SYINDX    1-105      3-86#
SYNAME    1-106      4-108#
SYNCH     1-132      3-82
SYPSPR    1-47       4-210#
SYPSWD    1-47      16-404     16-404     16-404#
SYSCHN    1-95       3-42#      3-202
SYSDAT    1-98       3-63#      3-205
SYSDMP    4-171     16-153#
SYSGEN    1-105      3-89#
SYSPSS   10-177#    16-404#    22-160
SYSUPD    1-99       3-73#
SYSVER    1-99       3-72#
SYTIMH    1-101      4-47#
SYTIML    1-101      4-48#
SYUNIT    1-99       3-71#
T        18-22      18-22#     18-29      18-29#     18-36      18-36#     20-69      20-69      20-69#     20-69#     21-25      21-25
         21-25#     21-25#
TBR       1-57      18-10      18-10#     18-17      18-25      18-32
TECO      4-350#
TIMOUT    4-22      16-412#
TIOBAS    1-82       4-70#
TK1CNT    1-65       4-51#
TK1SEC    1-65       4-49#
TK1VAL    1-103      4-50#
TK3SVL    1-111      4-53#
TK5VAL    1-111      4-52#
TMIDLH    1-107      4-225#
TMIDLL    1-107      4-226#
TMIOH     1-65       4-219#
TMIOL     1-65       4-220#
TMIOWH    1-66       4-223#
TMIOWL    1-66       4-224#
TMSWPH    1-66       4-221#
TMSWPL    1-66       4-222#
TMSWTH    1-107      4-217#
TMSWTL    1-107      4-218#
TMTOTH    1-106      4-213#
TMTOTL    1-106      4-214#
TMUSRH    1-106      4-215#
TMUSRL    1-106      4-216#
TNHL      1-49       18-10      18-10      18-10      18-10      18-10      18-10      18-10      18-10      18-10      18-10      18-10      18-10
         18-10      18-10      18-10      18-10      18-10      18-10      18-10      18-10      18-10      18-10      18-10      18-10
         18-10      18-10      18-10      18-10      18-10      18-10#
TOTON     1-93       4-200#
TSDEFS    1-134      4-107
TSDHIO   22-34      22-37#
TSGEN     1-42#      1-48
TSLICH    4-186     16-436#
TSR       1-56       18-10      18-10#     18-17      18-25      18-32
TSXSIT    1-69       4-54#
TSXVEC    3-8#       3-37
TSXVRS    1-47       2-4#       3-21
TTCSCH    1-56       2-15#
TTOP      3-112#     3-208
```

Cross reference table (CREF V05.04)

```
TTOPTS    1-113     3-103     3-111#
U$CL      4-177    16-187#
UBUSMP    1-87      4-203#
UCLBLK    1-115     4-110#
UCLDAT    1-115    16-21#
UCLMNC    4-18     16-194#
UCLNAM    1-52      4-109#
UCLORD    4-178    16-205#
UKMNAM    1-54      4-111#
UMRBAS   20-85     20-96#
UMREND   20-85     20-105#
UMRWHD   20-85     20-86#
UMSYTP    1-89      4-105#
UNIBUS    1-114     4-301#   20-97
USPLCH    1-87      2-85#
USRBAS    1-80      4-65#
USROFF    3-68#     3-207
USWPCH    1-119     2-87#     3-193
UXIFLG    4-176    16-141#
VBUSTP    1-114     4-174#
VCBBAS   22-6      22-7#
VCBEND   22-6      22-12#
VCORTM    1-83      4-11#
VCSHNB    1-66      4-37#
VCXCTL    1-75      4-190#
VCXTRM    1-75      4-189#
VDBFLG    1-118     4-181#
VDFMEM    1-108     4-29#
VDISPC    1-70      4-45#
VDMKTP    1-70      4-172#
VDMTCR    1-70      4-115#
VDOSPC    1-70      4-46#
VDSKBU    1-72      4-41#
VECBAS    1-108     3-37#     3-197     3-202     3-203     3-204     3-205     3-206     3-207     3-208     3-209     3-210
          3-211     3-212     3-213
VEDIT     1-64      4-191#
VERR     18-17     18-17#    18-25     18-25#    18-32     18-32#
VH$BA1    1-77      4-283#
VH$BA2    1-77      4-284#
VH$BCR    1-77      4-285#
VH$CSR    1-76      4-278#
VH$DBR    1-76      4-279#
VH$LCR    1-77      4-282#
VH$LPR    1-76      4-280#
VH$LSR    1-76      4-281#
VHIMEM    1-103     4-28#
VHIPCT    1-94      4-12#
VHOINT   22-35     22-48#
VHSTOP   22-35     22-45#
VHSTRT   22-35     22-44#
VHXOFF   22-36     22-47#
VHXON    22-36     22-46#
VIDCSR    1-100     4-114#
VINABT    1-63      4-175#
VINTIO    1-89      4-13#
VKEYMX    1-72      4-35#
```

Cross reference table (CREF V05.04)

```
VLDSYS    1-60       4-179#
VLSWCH    4-187     16-442#
VMAXMC    1-80       4-24#
VMIOBF    1-81       4-192#
VMIOSZ    1-79       4-193#
VMLBLK    1-90       4-17#
VMSCHR    1-79       4-25#
VMXCSH    1-85       4-94#
VMXFIL    1-108      4-19#
VMXMON    1-61       4-21#
VMXMRB    1-80       4-27#
VMXMSG    1-80       4-26#
VMXSF     1-90       4-14#
VMXSFC    1-90       4-15#
VMXWIN    1-55       4-34#
VNCSLO    1-71       4-42#
VNCXOF    1-71       4-43#
VNCXON    1-71       4-44#
VNFCSH    1-85       4-20#
VNGR      1-51       4-32#
VNRFLG    1-69       4-39#
VNUIP     1-51       4-36#
VNUMDC    1-81       4-16#
VOFFTM    1-86       4-23#
VPLAS     1-90       4-31#
VPRIDF    1-99       4-184#
VPRIHI    1-98       4-183#
VPRILO    1-98       4-182#
VPRIVR    1-64       4-185#
VQUAN0    1-73       4-4#
VQUAN1    1-83       4-5#
VQUAN2    1-83       4-9#
VQUAN3    1-73       4-10#
VQUN1A    1-83       4-6#
VQUN1B    1-83       4-7#
VQUN1C    1-69       4-8#
VSCHED    1-72       4-40#
VSLEDT    1-56       4-180#
VSWPFL    1-92       4-173#
VSWPSL    1-55       4-30#
VSYDMP    1-70       4-171#
VT100     1-133
VT200     1-133
VT52      1-133
VTMOUT    1-87       4-22#
VTSLCH    1-73       4-186#
VU$CL     1-52       4-177#
VUCLMC    1-115      4-18#
VUCLOR    1-115      4-178#
VUSPHN    1-126      4-194#
VUXIFL    1-62       4-176#
VVLSCH    1-73       4-187#
VVPWCH    1-73       4-188#
WILDFL    1-64      16-496#
WINBAS    1-80       4-67#
X       16-543     16-543#  16-544     16-544#  16-545     16-545#  16-546     16-546#  16-547     16-547#  16-548     16-548#
        16-549     16-549#  16-550     16-550#  16-551     16-551   16-551     16-551   16-551#    16-551#  16-551#    16-551#
```

Cross reference table (CREF V05.04)

```
BUFSIZ    9-49#    20-69    20-69
CHNRES    3-26#     3-43     3-44     3-45     3-46     3-47     3-48     3-49     3-50     3-51     3-52     3-53
          3-54     3-55     3-56     3-57     3-58     3-59    3-185    3-185    3-185    3-185    3-185    3-185
         3-185    3-185    3-189
CLDEF     8-11#
CLEND    10-132#
CLTABL   22-82#    22-93    22-94    22-95    22-96    22-97    22-98    22-99   22-100   22-101   22-102   22-103
         22-104   22-105   22-106   22-107   22-108   22-109   22-110   22-111
CMDFIL    9-83#    18-22    18-29    18-36
DETACH   10-141#   20-51    20-51
DEVBEG    5-7#    16-542
DEVDEF    5-40#   16-543   16-544   16-545   16-546   16-547   16-548   16-549   16-550   16-551   16-551   16-551
         16-551
DEVEND    5-109#  16-551
DHDEF    11-14#
DHUDEF   12-57#
DHVDEF   12-12#
DZDEF    13-44#
FLAGS     9-6#    18-21    18-28    18-35
LINDEF    8-58#   18-17    18-25    18-32
LINEND   10-8#    18-22    18-29    18-36    20-69    20-69
LINPRM    9-106#
MEMORY    4-307#   4-324
MUXDEF   13-12#
MUXEND   13-52#
MXTBL     4-241#   4-251    4-252    4-253    4-254    4-255    4-256    4-257    4-258    4-259    4-260    4-261
          4-262    4-263    4-264    4-265    4-266
NAME      9-29#   18-18    18-29    18-36
OB        6-7#    18-10    18-10    18-10    18-10    18-10    18-10    18-10    18-10    18-10    18-10    18-10
         18-10    18-10    18-10    18-10    18-10    18-10    18-10    18-10    18-10    18-10    18-10    18-10
         18-10    18-10    18-10    18-10    18-10    18-10    18-10    18-10    18-10    18-10    18-10    18-10
         18-10    18-10    18-10    18-10    18-10    18-10    18-10    18-10    18-10    18-10    18-10    18-10
         18-10    18-10    18-10    18-10    18-10    18-10    18-10    18-10    18-10    18-10    18-10    18-10
         18-10    18-10    18-10    18-10    18-10    18-10    18-10    18-10    18-10    18-10    18-10    18-10
         18-10
OBH       6-37#   18-10    18-10    18-10    18-10    18-10    18-10    18-10    18-10    18-10    18-10    18-10
         18-10    18-10    18-10    18-10    18-10    18-10    18-10    18-10    18-10    18-10    18-10    18-10
         18-10    18-10
OBP       6-22#   18-10    18-10    18-10    18-10    18-10    18-10    18-10
OBT       6-52#   18-10    18-10    18-10    18-10    18-10    18-10    18-10    18-10    18-10
PAGE      9-73#
RTDEF    10-198#  21-25    21-25
SILO      9-63#
SPEED     9-139#
SPOOL    14-13#   16-598
SRCHK     9-198#  18-17    18-25    18-32
SYSPS    10-178#  16-404
TBLDEF    6-71#   18-10
TRMTYP    9-12#
UMRDEF   20-87#
VECCHK    9-183#  18-17    18-25    18-32
```