```
                        ;***COPYRIGHT 1969, DIGITAL EQUIPMENT CORP., MAYNARD, MASS.***


                        ;THIS SUB-PROGRAM ASSEMBLED WITH SYSTEM PARAMETER FILE - S.MAC(V414)
                                XLIST
                                LIST
                        ;THIS SUB-PROGRAM ASSEMBLED WITH CONFIGURATION DEPENDENT FEATURE SWITCHES - FT50SB,MAC(
                        V003)
                                XLIST
                                LIST
                        TITLE   DTCSRN - NEW FORMAT DECTAPE SERVICE FOR 551 (PDP-6)
                        SUBTTL  DTA551 A.WACHS/TW/RCC 01 JUN 69  V012
                                XP      VDTASR,012      ; GLOBAL VERSION NUMBER FOR LOADER STORAGE MAP.


                                ENTRY DTCSRN    ;ENTRY POINT FOR SELECTIVE LOAD BY BUILD
000000                  DTCSRN:
                        EXTERNAL        TPOPJ,TPOPJ1,DTCCHL,DCOUT,DCIN,DCON,DCOFF
                        EXTERNAL        STOIOS,STOTAC,SETACT,CLRACT,OUT,DTCSAV,PIOMOD

                        EXTERNAL        DTAVAL,DTREQ,SETIOD,THSDAT,PUNIT,GETDCDT
                        EXTERNAL        ADVBFE,ADVBFF,ADPERR,WAIT1,CPOPJ,CPOPJ1,RADDIR
                        EXTERNAL        COMCHK,PJOBN,RELEA9,UADCK1,DTTRY
                        EXTERNAL        JBTADR,DCLOC,DCLOC1,DTCCHN,DCREQ,DCAVAL


                000004  BLK=4


                000144  DIRBLK=:D100    ;NUMBER OF BLOCK FOR DIRECTORY

                001101  TOPBLK=1101      ;HIGHEST LEGAL BLOCK NUMBER
                000123  NAMSTR=:D83      ;1ST NAME WORD IN DIRECTORY
                000005  QUANT=5          ;NUMBER OF BLOCKS CAN READ BEFORE GIVING UP DTC
                000006  SPACE=6          ;NUMBER OF BLOCKS SEPERATING CONTIGUOUS BLKS OF A FILE


                        ;DDB MAGIC CELLS
                000013  FSTBLK=13
                000014  DLOC=14
                000015  IBLK=15
                000016  OBLK=16
                000017  DISPAD=17
                000020  DMPLST=20
                000021  SVDWRD=21
```

```
                                ;FLAGS IN RH OF IOS
                    000100      UDSD=100
                                ;FLAGS IN LH OF IOS
                    000200      NOLINK=200
                    000400      CHNGDR=400
                    001000      RVERSE=1000
                    002000      SINGL=2000                  ;JUST READ OR WRITE 1 BLOCK
                    004000      DMPMOD=4000
                    010000      RWDIR=10000
                    020000      DMPCLS=20000
                    040000      NORUF=40000                 ;I/O DIRECTLY TO USER
                    737777      NORUFC=737777               ;-NORUF

          777777    777777      CPRIT=-1                    ;CONDITIONAL ASSEMBLY PARAMETER FOR I/O DIRECTLY
                                                            ;TO USER.  IF -1 THE I/O IN DUMP MODE WITH
                                                            ;BIT 29 ON IN INIT WILL DO IO DIRECTLY TO USER
                                                            ;WITHOUT DIRECT CONSIDERATION OF BLOCK BOUNDRIES
```

```
                                        INTERN  DTCINT,DTCDDB,DTCINI,DTCDDS

000000  446441  200000  DTCDDB: SIXBIT  /DTA0/
000001  000074  000200          XWD     .D60*HUNGST,200
000002  000000  000000          0
000003  000000  000224'          EXP     DTCDSP
000004  001107  154403          XWD     1107,154403
000005  000000  000000          EXP     0,0
000006  000000  000000
000007  000007  000000          XWD     PROG,0
000010  000007  000000          XWD     PROG,0
000011  000000  000000          EXP     0,0,0
000012  000000  000000
000013  000000  000000
000014  000000  000022'          EXP     DTCDIR
000015  000000  000000          EXP     0,0,0,0,0
000016  000000  000000
000017  000000  000000
000020  000000  000000

000021  000000  000000
000022                  DTCDIR: BLOCK   200
                000222  DTCDDS=.-DTCDDB


000222  254000  000243'          JRST    DTCINI
000223  254000  001445'          JRST    THRUTP          ;HUNG DEVICE
000224  254000  000677' DTCDSP: JRST    UREL
000225  254000  000663'          JRST    UCLS
000226  254000  000777'          JRST    UOUT
000227  254000  000730'          JRST    UIN
000230  254000  000472'          JRST    ENTR
000231  254000  000247'          JRST    LOOK
000232  254000  000773'          JRST    OMPO
000233  254000  000725'          JRST    OMPI
000234  254000  000576'          JRST    SETO
000235  254000  000574'          JRST    SETI
000236  254000  000622'          JRST    GETF
000237  254000  000315'          JRST    RENAM
000240  263140  000000          POPJ    PDP,            ;CLOSE INPUT
000241  254000  000642'          JRST    UTPCLR
000242  263140  000000          POPJ    PDP,            ;MTAPE


                        ;INITIALIZE DTC
000243  721200  000000  DTCINI: CONO    DTC,0
000244  720200  000000          CONO    DC,0
000245  513000  001357'          HLLZS   DTCINT  ;CLEAR CONSO
000246  263140  000000          POPJ    PDP,
```

```
                                        EXTERN  JOBSAV
                              ;LOOKUP A DIRECTORY ENTRY
000247  602000  000170  LOOK:  TRNE    IOS,UDSD        ;NON-STANDARD?
000250  254000  000000        JRST    CPOPJ1          ;YES, LOOKUP OK
000251  260140  000356'        PUSHJ   PDP,DSERCH      ;NO, FIND DIRECTORY ENTRY
000252  263140  000000        POPJ    PDP,            ;NOT THERE
000253  550101  000026        HRRZ    TAC1,26(TAC)    ;GET DATE, NO, OF 1K BLOCKS NEEDED
000254  350000  000014        AOS     UUO             ;POINT UUO TO WORD 3
000255  202120  000014        MOVEM   TAC1,@UUO       ;INTO USER'S LOOKUP BLOCK
000256  574040  000001  LOOKA: HLRE    TAC,TAC         ;GET INDEX
000257  271040  000027        ADDI    TAC,27
000260  336006  000013        SKIPN   FSTBLK(DEVDAT)  ;TAPE BEING WRITTEN?
000261  506046  000016        HRLM    TAC,OBLK(DEVDAT) ;NO, SAVE INDEX FOR POSSIBLE OUTPUT
000262  542040  000015        HRRM    TAC,AC1         ;SAVE INDEX IN CASE LH(IBLK) IS
                                                      ;CHANGED BY DEAD-RECKONING
000263  201100  000000        MOVEI   TAC1,0
000264  260140  000415'        PUSHJ   PDP,BLKSRC      ;COUNT NUMBER OF BLOCKS IN FILE
000265  344600  000271'        AOJA    UUO,STOWD4      ;LAST BLOCK DONE
000266  350000  000002        AOS     TAC1            ;COUNT THE BLOCK
000267  370003  000001        SOS     1(PDP)          ;ADJUST FOR CORRECT RETURN
000270  253140  000423'        AOBJN   PDP,BLKSRB      ;LOOK FOR NEXT BLOCK

                              ;TAC1 HAS THE NUMBER OF BLOCKS BELONGING TO THE FILE
000271  220170  002077' STOWD4: IMUL    TAC1,[-177]     ;-NUMBER OF WORDS IF ALL BLOCKS FULL
000272  516120  000014        HRLZM   TAC1,@UUO       ;STORE IN DIRECTORY WD 4
000273  275600  000002        SUBI    UUO,2           ;POINT UUO TO DIRECTORY WD 2
000274  621000  000000        TLZ     IOS,IO          ;MAKE SURE IO IS OFF
000275  661000  002000        TLO     IOS,SINGL       ;JUST READ 1 RECORD
000276  201200  000144        MOVEI   BLK,DIRBLK      ;NO, FIND FIRST MENTION OF BLOCK
000277  260140  000443'        PUSHJ   PDP,LSTFRE+1    ;NEAR DIRECTORY
000300  326200  000303'        JUMPN   BLK,.+3         ;FOUND IF BLK NOT =0
000301  260140  000415' LOOKD: PUSHJ   PDP,BLKSRC      ;FIND FIRST MENTION IN DIRECTORY
000302  254000  001652'        JRST    BDDIR           ;NOT THERE - ERROR
000303  260140  001176'        PUSHJ   PDP,READRF      ;GO READ IT
000304  260140  000000        PUSHJ   PDP,WAIT1       ;WAIT TILL IT'S IN
000305  506646  000015        HRLM    AC1,IBLK(DEVDAT) ;SAVE INDEX ON INPUT FILE FOR LATER
                                                      ;TEST ON ENTER - WONT ALLOW ENTER
                                                      ;TO BE DONE ON LOOKED-UP FILE
000306  550206  000015        HRRZ    BLK,IBLK(DEVDAT) ;GET FIRST BLOCK OF FILE
000307  336006  000013  LOOKC: SKIPN   FSTBLK(DEVDAT)  ;IF FILE HAS NOT BEEN ENTERED
000310  202206  000013        MOVEM   BLK,FSTBLK(DEVDAT) ;SAVE IN DDB
000311  542200  000014        HRRM    BLK,@UUO        ;SAVE IN USER'S AREA
000312  500060  000014        HLL     TAC,@UUO        ;GET USER'S EXTENSION
000313  502046  000012        HLLM    TAC,DEVEXT(DEVDAT)      ;SAVE IN DEVICE DATA BLOCK FOR RENAME AND
                                                      ; AND SUPERSEDING SHARED SEGMENTS
000314  254070  000250'        JRST    CPOPJ1          ;AND TAKE GOOD EXIT
```

```
  152  000315  261140  000014  RENAM:  PUSH    P,UUO           ;SAVE LOC OF NEW NAME
  153  000316  201076  000011          MOVEI   UUO,DEVFIL(DEVDAT) ;SEARCH FOR OLD NAME
  154  000317  260140  000360'         PUSHJ   P,DSER1
  155  000320  254070  000350'         JRST    RENER1          ;NOT FOUND - ERROR
  156  000321  262140  000014          POP     P,UUO           ;FOUND, RESTORE UUO
  157  000322  332020  000014          SKIPE   @UUO            ;RENAMING TO ZERO?
  158  000323  254000  000335'         JRST    RENAM2          ;NO, GO DO REAL RENAME
  159  000324  402001  000000          SETZM   (TAC)           ;YES, DELETE NAME IN DIR
  160  000325  402041  000026          SETZM   26(TAC)         ;DELETE EXTENSION
  161  000326  402076  000011          SETZM   DEVFIL(DEVDAT)  ;ZERO DEVFIL
  162  000327  574040  000001          HLRE    TAC,TAC         ;GET INDEX OF FILE
  163  000330  271040  000027          ADDI    TAC,27
  164  000331  260140  000464'         PUSHJ   P,DLETE         ;DELETE ALL BLOCKS OF FILE
  165  000332  661000  000400  RENAM1: TLO     IOS,CHNGDR      ;DIRECTORY HAS CHANGED
  166  000333  350003  000000          AOS     (P)             ;SET FOR GOOD RETURN
  167  000334  254000  000000          JRST    STOIOS          ;GO TO USER
  168
  169                                ;COME HERE TO RENAME TO A REAL NEW NAME
  170  000335  200240  000001  RENAM2: MOVE    DAT,TAC          ;SAVE LOC OF NAME IN DIRECTORY

  171  000336  260140  000356'         PUSHJ   P,DSERCH        ;SEARCH FOR NEW NAME
  172  000337  334000  000000          SKIPA                   ;NOT FOUND - GOOD
  173  000340  254000  000352'         JRST    RENER2          ;NAME ALREADY EXISTS - ERROR
  174  000341  200060  000014          MOVE    TAC,@UUO        ;GET NEW NAME
  175  000342  202045  000000          MOVEM   TAC,(DAT)       ;SAVE IN DIR
  176  000343  350000  000014          AOS     UUO
  177  000344  200060  000014          MOVE    TAC,@UUO        ;EXTENSION
  178  000345  502045  000026          HLLM    TAC,26(DAT)     ;SAVE IN DIR
  179  000346  502046  000012          HLLM    TAC,DEVEXT(DEVDAT) ;SAVE INN DDB
  180  000347  254070  000332'         JRST    RENAM1          ;GIVE GOOD RETURN TO USER
  181
  182  000350  262140  000014  RENER1: POP     P,UUO
  183  000351  634040  000001          TDZA    TAC,TAC         ;RH E+1 =0
  184  000352  201040  000004  RENER2: MOVEI   TAC,4           ;RH E+1 =1
  185  000353  350000  000014          AOS     UUO             ;POINT TO 2ND WORD
  186  000354  542060  000014          HRRM    TAC,@UUO        ;SET ERRORR CODE
  187  000355  263140  000000          POPJ    P,              ;AND TAKE ERROR RETURN
```

```
                                ;SEARCH DIRECTORY FOR A MATCH
000356  201654  000003  DSERCH: MOVEI   AC1,3(UUO)       ;CHECK VALIDITY OF ADDRESS
000357  260140  007000          PUSHJ   PDP,UADCK1        ;NEVER RETURN IF ADDR. OUT OF BOUNDS
000360  260140  000471' DSER1:  PUSHJ   PDP,DIRCHK       ;ENSURE DIRECTORY IS IN CORE
000361  550046  000014          HRRZ    TAC,DLOC(DEVDAT) ;LOCATION OF DIRECTORY
000362  270040  002100'          ADD     TAC,EXWD -26,NAMSTR) ;POINT TO START OF NAMES
000363  336120  000014  NMLOOK: SKIPN   TAC1,@UUO         ;GET NAME
000364  254000  000000          JRST    TPOPJ             ;NULL ARGUMENT - ERROR RETURN
000365  202106  000011          MOVEM   TAC1,DEVFIL(DEVDAT)     ;STORE FOR RENAME AND SUPERSEDING
                                                          ; SHARED SEGMENTS
000366  316101  000000          CAMN    TAC1,(TAC)        ;TEST FOR MATCH
000367  344600  000372'          AOJA    UUO,NMFOUN        ;FOUND NAME, CHECK EXTENSION
000370  253040  000366'          AOBJN   TAC,.-2           ;TRY NEXT NAME
000371  263140  000000          POPJ    PDP,              ;NOT FOUND
000372  510120  000014  NMFOUN: HLLZ    TAC1,@UUO         ;PICK UP USER'S EXTENSION
000373  430101  000026          XOR     TAC1,26(TAC)      ;TEST AGAINST DIRECTORY EXTENSION
000374  607100  777777          TLNN    TAC1,-1           ;MATCH?
000375  254000  000314'          JRST    CPOPJ1            ;YES, RETURN
000376  252040  000400'          AOBJP   TAC,.+2
000377  364600  000363'          SOJA    UUO,NMLOOK        ;NO, TRY NEXT NAME
000400  364600  000000          SOJA    UUO,CPOPJ         ;NAME NOT FOUND

                                ;CHECK IF DIRECTORY IS IN CORE, IF NOT, READ IT
000401  606000  000100  DIRCHK: TRNN    IOS,UDSD          ;DONT BOTHER IF NON-STANDARD
000402  337036  000004          SKIPG   DEVMOD(DEVDAT)    ;IS IT IN?
000403  263140  000000          POPJ    PDP,              ;YES, RETURN
000404  201700  000144          MOVEI   RLK,DIRBLK        ;BLOCK NUMBER
000405  621000  000020          TLZ     IOS,IO
000406  260140  000712'          PUSHJ   PDP,GETDT         ;GET CONTROL
000407  661000  010000          TLO     IOS,RWDIR         ;JUST READ 1 BLOCK
000410  260140  001200'          PUSHJ   PDP,READBC        ;GO READ IT
000411  260140  000374'          PUSHJ   PDP,WAIT1         ;WAIT TILL IN
000412  205100  400000          MOVSI   TAC1,DVDIRI       ;SET DIRECTORY-IN-CORE BIT
000413  436136  000004          ORM     TAC1,DEVMOD(DEVDAT)
000414  263140  000000          POPJ    PDP,
```

```
                                ;SEARCH DIRECTORY FOR FILE WHOSE INDEX IS IN TAC
000415  205240  440500  BLKSRC: MOVSI   DAT,440500       ;DAT IS A BLOCK POINTER
000416  540246  000014          HRR     DAT,DLOC(DEVDAT)
000417  201200  000001          MOVEI   BLK,1            ;START AT BLOCK 1

000420  134400  000005  BLKSRA: ILDB    TEM,DAT          ;INDEX OF NEXT BLOCK
000421  316040  000010          CAMN    TAC,TEM          ;MATCH?
000422  254000  000375'         JRST    CPOPJ1           ;YES, RETURN

000423  305200  001101  BLKSRB: CAIGE   BLK,TOPBLK       ;NO, SEARCHED LAST?
000424  344200  000420'         AOJA    BLK,BLKSRA       ;NO, TRY NEXT BLOCK
000425  263140  000000          POPJ    PDP,             ;YES, RETURN

                                ;SET UP POINTER TO DIRECTORY FOR BLOCK IN BLK
000426  261140  000004  SETPTR: PUSH    PDP,BLK          ;SAVE BLK
000427  260140  000543'         PUSHJ   PDP,DRPTR        ;SET BLK AS A BYTE POINTER
000430  200240  000004          MOVE    DAT,BLK          ;RETURN IT IN DAT
000431  262140  000004          POP     PDP,BLK          ;RESTORE BLK
000432  263140  000000          POPJ    PDP,             ;AND RETURN

                                ;GET NEXT AVAILABLE FREE BLOCK
000433  261140  000010  NXTFRE: PUSH    PDP,TEM
000434  260140  000426'         PUSHJ   PDP,SETPTR       ;SET DAT TO A BYTE POINTER
000435  201040  000000          MOVEI   TAC,0            ;LOOK FOR FREE BLOCKS
000436  260140  000420'         PUSHJ   PDP,BLKSRA       ;FIND A ZERO BLOCK
000437  201200  000000          MOVEI   BLK,0            ;NOT THERE- RETURN 0
000440  262140  000010  FREXIT: POP     PDP,TEM
000441  263140  000000          POPJ    PDP,

                                ;GET PREVIOUS FREE BLOCK
000442  201040  000000  LSTFRE: MOVEI   TAC,0
000443  261140  000010          PUSH    PDP,TEM
000444  271200  000002          ADDI    BLK,2
000445  260140  000426'         PUSHJ   PDP,SETPTR       ;SET DAT AS A POINTER
000446  275200  000002          SUBI    BLK,2
000447  260140  000455'         PUSHJ   PDP,DECPTR       ;DECREMENT BYTE POINTER
000450  135400  000005          LDB     TEM,DAT          ;INDEX TO BLOCK
000451  316400  000001          CAMN    TEM,TAC          ;FOUND?
000452  254000  000440'         JRST    FREXIT           ;YES, RETURN
000453  367200  000447'         SOJG    BLK,.-4          ;TRY AGAIN IF NOT AT START
000454  254000  000440'         JRST    FREXIT           ;REACHED START - RETURN BLK=0

                                ;DECREMENT BYTE POINTER
000455  321240  000462' DECPTR: JUMPL   DAT,.+5
000456  270240  002101'         ADD     DAT,[BYTE (6) 5] ;DECREMENT
000457  327240  000400'         JUMPG   DAT,CPOPJ        ;IF POSITIVE - SAME WORD
000460  505240  010500          HRLI    DAT,010500       ;RESET TO PREVIOS WORD
000461  364240  000457'         SOJA    DAT,CPOPJ
000462  505240  060500          HRLI    DAT,060500
000463  364240  000461'         SOJA    DAT,CPOPJ
```

```
                                        ;COME HERE TO DELETE THE FILE WHOSE INDEX IS INN TAC
000464  201100  000000  DLETE:  MOVEI   TAC1,0           ;SET TO DELETE BLOCKS
000465  260140  000415'         PUSHJ   PDP,BLKSRC       ;FIND A BLOCK BELONGING TO FILE
000466  254000  000463'         JRST    CPOPJ    ;ALL THROUGH
000467  137100  000035          DPB     TAC1,DAT         ;DELETE IT
000470  370003  000001          SOS     1(PDP)           ;ADJUST PDL FOR RETURN
000471  253140  000423'         AOBJN   PDP,BLKSRB       ;AND FIND NEXT MATCH


                                        ;ENTER A FILE NAME IN DIRECTORY
000472  602000  000100  ENTR:   TRNE    IOS,UDSD         ;NON STANDARD?
000473  254020  000422'         JRST    CPOPJ1           ;YES, RETURN
000474  260140  000356'         PUSHJ   PDP,DSERCH       ;NO, LOOK FOR MATCH
000475  254000  000553'         JRST    NEWENT           ;THIS IS A NEW ENTRY
000476  200120  000014  ENTR2:  MOVE    TAC1,@UUO        ;PICK UP 2ND WORD (EXTENSSION)
000477  350020  000014          AOS     UUO              ;POINT TO WORD 3
000500  540120  000014          HRR     TAC1,@UUO        ;ADD DATE
000501  606100  007777          TRNN    TAC1,7777        ;IS DATE ALREADY THERE?
000502  434100  000000          IOR     TAC1,THSDAT      ;NO, ADD CURRENT DATE
000503  202101  000026          MOVEM   TAC1,26(TAC)     ;INTO DIRECTORY
000504  335746  000016          SKIPGE  AC3,OBLK(DEVDAT) ;IS THIS A SAVE FILE (UGETF DONE
                                                         ;BEFORE THE ENTER?)
000505  344020  000560'         AOJA    UUO,SETWD4       ;YES, STORE LENGTH IN DIRECTORY
000506  275000  000002  ENTRA:  SUBI    UUO,2            ;NO, POINT TO NAME
000507  200120  000014          MOVE    TAC1,@UUO        ;PICK IT UP
000510  202101  000000          MOVEM   TAC1,(TAC)       ;INTO DIRECTORY
000511  574040  000001          HLRE    TAC,TAC          ;COMPUTE INDEX OF FILE
000512  271040  000027          ADDI    TAC,27
000513  554246  000015          HLRZ    DAT,IBLK(DEVDAT) ;INDEX OF INPUT FILE
000514  274240  000001          SUB     DAT,TAC          ;WRITING SAME FILE AS READING?
000515  322240  000466'         JUMPE   DAT,CPOPJ        ;TAKE ERROR RETURN IF YES
000516  506046  000016          HRLM    TAC,OBLK(DEVDAT) ;SAVE INDEX IN DDB

000517  260140  000464'         PUSHJ   PDP,DLETE        ;DELETE ALL BLOCKS BELONGING TO FILE
000520  342740  000570'         AOJE    AC3,ENTRD        ;FIND FIRST FREE BLOCK ON TAPE IF THIS
                                                         ;IS A SAVE FILE (UGETF DONE)
000521  201100  000144          MOVEI   BLK,DIRBLK       ;NO, GET 1ST BLOCK CLOSE TO
000522  661000  001000          TLO     IOS,RVERSE       ;DIRECTORY, GOING IN REVERSE
000523  260140  000636'         PUSHJ   PDP,USLSTA
000524  303200  001101          CAILE   BLK,TOPBLK       ;BLOCK LEGAL?
000525  263140  000000          POPJ    PDP,             ;NO, ERROR RETURN
000526  202206  000013  ENTRC:  MOVEM   BLK,FSTBLK(DEVDAT) ;SAVE AS 1ST BLOCK
000527  542206  000016          HRRM    BLK,OBLK(DEVDAT)         ;SAVE IN DDB
000530  350020  000014          AOS     UUO              ;POINT UUO TO WORD 2
000531  542220  000014          HRRM    BLK,@UUO         ;SAVE 1ST BLOCK IN USER'S AREA
000532  502060  000014          HLL     TAC,@UUO         ;GET EXTENSION
000533  502046  000012          HLLM    TAC,DEVEXT(DEVDAT)       ;SAVE EXTENSION IN DDB ALSO
000534  661000  000200          TLO     IOS,NOLINK
000535  350003  000000          AOS     (PDP)
```

```
                                ;MARK DIRECTORY ENTRY POINTED TO BY BLK AS TAKEN
000536  260140  000543' MARKDR: PUSHJ   P,DRPTR          ;SET POINTER TO BLOCK IN DIR
000537  554046  000016          HLRZ    T,@BLK(DEVDAT)  ;PICK UP INDEX
000540  136040  000004          IDPB    T,BLK           ;MARK DIRECTORY
000541  661000  000400          TLO     IOS,CHNGDR      ;DIRECTORY HAS CHANGED
000542  254000  000334'         JRST    STOIOS

                                ;;SET POINTER TO CORRECT DIRECTORY ENTRY
000543  275200  000001  DRPTR:  SUBI    BLK,1           ;SET FOR ILDB OR IDPB
000544  231200  000007          IDIVI   BLK,7           ;COMPUTE WORD, POSITION
000545  270200  000014          ADD     BLK,@LOC(DEVDAT) ;GET CORRECT ADDRESS
000546  505200  440500          HRLI    BLK,440500      ;MAKE IT A BYTE POINTER
000547  322240  000515'         JUMPE   DAT,CPOPJ       ;CORRECT FOR POSITION IN WORD
000550  133000  000004          IBP     BLK
000551  367240  000550'         SOJG    DAT,.-1
000552  263140  000000          POPJ    P,

                                ;HERE FOR NEW FILE NAME ON ENTER
000553  274040  002102' NEWENT: SUB     T,[XWD 26,26];START AT BEGINNING OF DIRECT.

000554  336001  000000          SKIPN   (T)             ;FIND A FREE SLOT
000555  344600  000476'         AOJA    U,UUO,ENTR2     ;RETURN WITH UUO POINTING TO WRD 2
000556  253040  000554'         AOBJN   T,.-2
000557  263140  000000          POPJ    P,              ;NONE AVAILABLE.

                                ;SET UP LENGTH OF FILE IN DIRECTORY FOR A SAVE FILE
000560  574120  000014  SETWD4: HLRE    T1,@UUO         ;GET -LENGTH
000561  213000  000002          MOVNS   T1              ;+LENGTH
000562  570420  000014          HRRE    TEM,@UUO
000563  270100  000010          ADD     T1,TEM          ; +START ADDRESS
000564  620100  001777          TRZ     T1,1777         ;STORE N-1, WHERE N IS NO OF K
000565  242100  000002          LSH     T1,2
000566  436101  000026          ORM     T1,26(T)        ;INTO 2ND WRD OF DIRECTORY
000567  364600  000506'         SOJA    U,UUO,ENTRA     ;CONTINUE WITH ENTER

000570  201040  000000  ENTRD:  MOVEI   T,0             ;GET THE 1ST FREE BLOCK ON TAPE
000571  260140  000415'         PUSHJ   P,BLKSRC        ;AS THE 1ST LOGICAL BLOCK OF THE FILE
000572  263140  000000          POPJ    P,              ;NONE AVAILABLE
000573  254000  000526'         JRST    ENTRC           ;CONTINUE WITH ENTER
```

```
                                ;USETI  -  SET NEXT INPUT BLOCK TO READ
000574  630000  002133' SETI:    TDZ     IOS,[XWD IOEND,IODEND]
000575  334240  000006          SKIPA   DAT,DEVDAT

                                ;USETO  -  SET NEXT OUTPUT BLOCK TO READ
000576  201246  000001  SETO:    MOVEI   DAT,1(DEVDAT)
000577  260140  000411'          PUSHJ   PDP,WAIT1       ;WAIT FOR BUFFERS TO FILL (OR EMPTY)
000600  542665  000015          HRRM    UUO,IBLK(DAT)   ;SET BLOCK NUMBER
000601  254000  000542'          JRST    STOIOS          ;STOE IOS, POPJ

                                ;UGETF  -  GET NEXT FREE BLOCK FOR THIS FILE
000602  260140  000577' GETF:    PUSHJ   PDP,WAIT1       ;WAIT TILL BUFFERS EMPTY
000603  260140  000421'          PUSHJ   PDP,DIRCHK      ;ENSURE DIR, IN CORE
000604  260140  000611'          PUSHJ   PDP,USRFRE      ;GET NEXT AVAILABLE BLOCK
000605  336006  000016          SKIPN   OBLK(DEVDAT)    ;IF AN ENTER HAS NOT BEEN DONE,
000606  477276  000016          SETOB   BLK,OBLK(DEVDAT) ;SET SWITCH SO THAT ENTER WILL GET
                                                        ;1ST FREE BLOCK ON TAPE (FOR IODMP)
000607  200040  000004          MOVE    TAC,BLK         ;TELL USER THE BLOCK NUMBER
000610  254070  000000          JRST    STOTAC


                                ;GET NEXT (OR PREVIOUS) FREE BLOCK
000611  201440  000006  USRFRE: MOVEI   TEM,SPACE       ;BLOCKS "SPACE" APART
000612  135270  000000          LDB     BLK,PIOMOD      ;EXCEPT DUMP-MODE
000613  301200  000015          CAIL    BLK,5D
000614  201470  000002          MOVEI   TEM,2
000615  201470  000002          MOVEI   TEM,2           ;WHICH ARE CLOSER
000616  550276  000016  USRFRA: HRRZ    BLK,OBLK(DEVDAT)        ;CURRENT BLOCK
000617  603070  001200          TLNE    IOS,RVERSE      ;FORWARD?
000620  254070  000633'          JRST    USRLST          ;NO
000621  271210  000000          ADDI    BLK,(TEM)       ;YES, FIND NEXT BLOCK AT LEAST N
000622  303200  001171          CAILE   BLK,TOPBLK
000623  634220  000004          TDZA    BLK,BLK
000624  260140  000433' CALNXT: PUSHJ   PDP,NXTFRE       ;BLOCKS PAST THIS ONE
000625  326200  000601'          JUMPN   BLK,STOIOS      ;RETURN IF FOUND
000626  663400  000001          TLOE    TEM,1           ;FOUND NONE ON THIS PASS
000627  254000  000640'          JRST    NOBLKS          ;TAPE IS FULL
000630  641000  001000          TLC     IOS,RVERSE      ;REVERSE DIRECTION
000631  541400  000001          HRRI    TEM,1           ;START LOOKING AT NEXT BLOCK IN OTHER DIRECTION
000632  254000  000616'          JRST    USRFRA
000633  275210  000000  USRLST: SUBI    BLK,(TEM)       ;LOOK FOR FREE BLOCK N BEFORE
000634  337000  000004          SKIPG   BLK
000635  634270  000004          TDZA    BLK,BLK         ;REVERSE IF AT FRONT OF TAPE
000636  260140  000442' USLSTA: PUSHJ   PDP,LSTFRE       ;THIS ONE
000637  254000  000625'          JRST    CALNXT+1


                                ;NO FREE BLOCKS AVAILABLE, GIVE HIGH BLOCK,SET IOBKTL LATER
000640  201200  001102  NOBLKS: MOVEI   BLK,TOPBLK+1    ;SET HIGH BLOCK
000641  263140  000000          POPJ    PDP,
```

```
411                                     ;UTPCLR UUO
412  000642  602000  000100  UTPCLR:  TRNE    IOS,UDSD
413  000643  263140  000000           POPJ    PDP,            ;FORGET IT FOR NON-STANDARD
414  000644  205040  400000           MOVSI   TAC,400000      ;SET DIRECTORY-IN-CORE BIT
415  000645  436046  000004           ORM     TAC,DEVMOD(DEVDAT)
416  000646  661000  000400           TLO     IOS,CHNGDR      ;DIRECTORY HAS CHANGED
417  000647  550046  000014           HRRZ    TAC,DLOC(DEVDAT) ;LOC OF DIRECTORY
418  000650  504100  000001           HRL     TAC1,TAC
419  000651  541101  000001           HRRI    TAC1,1(TAC)     ;BLT POINTER
420  000652  402001  000000           SETZM   (TAC)
421  000653  251101  000176           BLT     TAC1,176(TAC)   ;LEAVE LAST WORD IN DIR, ALONE
422  000654  205100  017000           MOVSI   TAC1,17000      ;MARK DIRECTORY AS UNAVAILABLE
423  000655  202101  000016           MOVEM   TAC1,16(TAC)
424  000656  205100  757000           MOVSI   TAC1,757000     ;RESERVE BLOCKS 1 AND 2
425  000657  202101  000000           MOVEM   TAC1,(TAC)      ;FOR READ IN MODE LOADER
426  000660  205100  777770           MOVSI   TAC1,777770     ;MARK BLOCKS 1102-1105 AS
427  000661  456101  000122           ORCAM   TAC1,NAMSTR-1(TAC)  ;UNAVAILABLE ALSO
428  000662  254000  000625'          JRST    STOIOS
```

```
                                ;CLOSE UUO
000663  623000  000200  UCLS:   TLZE    IOS,NOLINK      ;IF NO LINK TO FREE BLOCK
000664  602000  000100          TRNE    IOS,UDSD        ;OR IF NON-STD MODE
000665  254000  000662'          JRST    STOIOS          ;FORGET IT
000666  135040  000612'          LDB     TAC,PIOMOD      ; WRITE LAST BLOCK
000667  301040  000016          CAIL    TAC,16          ;DUMP MODE?
000670  254000  000721'          JRST    CLSDMP          ;YES, CLOSE DUMP MODE
000671  201066  000010          MOVEI   TAC,@DEVOAD(DEVDAT) ;LOC OF BUFFER
000672  200101  000001          MOVE    TAC1,1(TAC)     ;LINK WORD
000673  667100  777777          TLON    TAC1,-1         ;LINK=-1 IF NOT SPECIFIED
000674  202101  000001          MOVEM   TAC1,1(TAC)     ;LINK = -1,,, EOF
000675  202006  000002          MOVEM   IOS,DEVIOS(DEVDAT)  ;SAVE IOS
000676  254000  000000          JRST    OUT             ;GO TO WRITE RECORD


                                ;RELEASE UUO
000677  402006  000013  UREL:   SETZM   FSTBLK(DEVDAT)  ;ZERO FSTBLK
000700  260140  001170'          PUSHJ   PDP,NXTCM2      ;CLEAR OUT DUMP-MODE STUFF
000701  620000  000100          TRZ     IOS,UDSD        ;CLEAR NON-STD BIT
000702  337006  000004          SKIPG   DEVMOD(DEVDAT)  ;IF DIRECTORY HAS BEEN
000703  627000  000400          TLZN    IOS,CHNGDR      ;MODIFIED IT MUSY BE WRITTEN
000704  263140  000000          POPJ    PDP,            ;NOT TOUCHED
000705  661000  010020          TLO     IOS,RWDIR+IO    ;GOING TO WRITE IT
000706  260140  000712'          PUSHJ   PDP,GETDT       ;WAIT TILL DTC AVAILABLE
000707  201200  000144          MOVEI   BLK,DIRBLK      ;BLOCK NUMBER
000710  260140  001045'          PUSHJ   PDP,WRTBLK      ;WRITE UT
000711  254000  000602'          JRST    WAIT1           ;DONT RETURN TO USER TILL DONE


                                ;GET DEC TAPE CONTROLLER
000712  260140  000000  GETDT:   PUSHJ   PDP,GETDCDT     ;GET DATA CONTROL, DECTAPE CONTROL
000713  352000  000000          AOSE    DTREQ           ;ARGUMENT FOR GETDCD
000714  201040  000005          MOVEI   TAC,QUANT       ;HAVE IT NOW
000715  202040  001664'          MOVEM   TAC,QUANTM      ;KEEP IT FOR QUANT BLOCKS
000716  202340  001657'          MOVEM   PROG,USEPRG
000717  202300  001656'          MOVEM   DEVDAT,USEWRD   ;SAVE ACS NEEDED ON INTERRUPT LEVEL
000720  254000  000000          JRST    SETACT          ;LIGHT IOACT AND RETURN


                                ;HERE TO CLOSE A DUMP MODDE FILE
000721  661000  000721' CLSDMP: TLO     IOS,CLSDMP      ;SET SWITCH
000722  260140  000712'          PUSHJ   PDP,GETDT       ;GET CONTROL
000723  402000  001677'          SETZM   BUF             ;ENSURE LINK, WORDCOUNT=0
000724  254000  001022'          JRST    DUFULL          ;GO WRITE LAST RECORD
```

```
                                        EXTERN  JOBDDT,USRDDT
                                ;DUMP MODE INPUT
000725                          DMPI:   IFN     CPBIT, <
000725  550736  000015                  HRRZ    AC2,IBLK(DEVDAT)
                                >
000726  260140  001070'                  PUSHJ   PDP,DMPSET      ;SET UP DUMP-MODE STUFF
000727  254000  000762'                  JRST    ZERCOR          ;ZERO USER'S CORE IF SAVE-MODE

                                ;INPUT UUO
000730  621000  000020          UIN:    TLZ     IOS,IO
000731  550736  000015                  HRRZ    BLK,IBLK(DEVDAT) ;BLOCK TO READ
000732  260140  000665'                  PUSHJ   PDP,STOIOS
000733  602000  000100                  TRNE    IOS,UDSD        ;NON STANDAR?
000734  254000  001176'                  JRST    READBF
000735  322200  000756'                  JUMPE   BLK,EOF         ;0 MEANS EOF
000736  260140  000752'                  PUSHJ   PDP,BLKCHK      ;CHECK LEGALITY OF BLOCK NUMBER
000737  607000  004000                  TLNN    IOS,DMPMOD      ;DUMP MODE?
000740  322200  000144                  CAIE    BLK,DIRBLK      ;TRYING TO READ DIRECTORY?
000741  254000  001176'                  JRST    READBF          ;NO. GO READ

                                ;READING DIRECTORY - GIVE CORE IMAGE IF IT EXISTS
000742  260140  000471'                  PUSHJ   PDP,DIRCHK      ;READ IT IF IT ISN'T IN ALREADY
000743  504046  000014                  HRL     TAC,DLOC(DEVDAT) ;LOC OF DIRECTORY
000744  201126  000027                  MOVEI   TAC1,@DEVIAD(DEVDAT)  ;WHERE USER WANTS IT
000745  541042  000001                  HRRI    TAC,1(TAC1)     ;LOC OF  DATA
000746  251042  000200                  ADJ     TAC,200(TAC1)   ;GIVE IT TO HIM
000747  260140  000020                  PUSHJ   PDP,ADVBFF      ;ADVANCE BUFFERS
000750  255000  000000                  JFCL
000751  263140  000000                  POPJ    PDP,
```

```
                                ;CHECK VALIDITY OF BLOCK NUMBER
000752  307200  001101  BLKCHK: CAIG    BLK,TOPBLK      ;LEGAL?
000753  263140  002000          POPJ    P0P,            ;YES, RETURN
000754  262140  000001          POP     P0P,TAC
000755  664000  040000          TROA    IOS,IOBKTL      ;NO, LIGHT ERROR BIT

                                ;INPUT BLOCK = 0 - END OF FILE
000756  661000  000040  EOF:    TLO     IOS,IOEND       ;LIGHT EOF BIT
000757  607000  004000          TLNN    IOS,DMPMOD
000760  254000  000732'          JRST    STOIOS
000761  254000  001612'          JRST    DMPEOF          ;GIVE UP CONTROL IF DUMP-MODE


                                ;ZERO USER'S CORE ON SAVE-MODE INPUT
000762  201047  000000  ZERCOR: MOVEI   TAC,JOBDDT(PROG) ;ZERO CORE
000763  505041  000001          HRLI    TAC,1(TAC)
000764  207000  000001          MOVSS   TAC             ;BLT POINTER
000765  550176  000220          HRRZ    TAC1,DMPLST(DEVDAT)  ;TOP CELL TO ZERO (-175)
000766  271177  000200          ADDI    TAC1,(PROG)     ;RELOCATE TO USER AREA
000767  402001  777777          SETZM   -1(TAC)         ;ZERO
000770  251042  000000          BLT     TAC,(TAC1)
000771  402000  000000          SETZM   USRDDT          ;DDT IS KEPT IN PROTECTED PART
000772  254000  000730'          JRST    WIN
```

```
                                ;DUMP MODE OUTPUT
000773  260140  000401' DMPO:   PUSHJ   PDP,DIRCHK       ;MAKE SURE DIRECTORY IS IN CORE
                               IFN     CPBIT, <
000774  550706  000016          HRRZ    AC2,OBLK(DEVDAT)
                               >
000775  260140  001070'          PUSHJ   PDP,DMPSET       ;SET DUMP-MODE POINTERS
000776  255000  000000          JFCL
                                ;OUTPUT UUO
000777  661000  000020  UOUT:   TLO     IOS,IO
001000  602000  000100          TRNE    IOS,UDSD         ;NON STANDARD?
001001  254000  001010'          JRST    UOUT2            ;YES
001002  260140  000401'          PUSHJ   PDP,DIRCHK       ;NO. MAKE SURE DIRECTORY IS IN CORE
001003  550706  000016          HRRZ    BLK,OBLK(DEVDAT)
001004  306700  000144          CAIN    BLK,DIRBLK       ;CHECK IF WRITING DIRECTORY
001005  254000  001052'          JRST    COR2HM           ;YES, WRITE CORE IMAGE
001006  322700  001062'          JUMPE   BLK,FAKADV       ;DONT WRITE IF NO BLOCK GIVEN
001007  260140  000752'          PUSHJ   PDP,BLKCHK       ;CHECK FOR LEGAL BLOCK
001010  607600  004000  UOUT2:   TLNN    IOS,DMPMOD       ;ALREADY HAVE CONTROL IF DUMP-MODE
001011  260140  000712'          PUSHJ   PDP,GETDT        ;GET DEC TAPE CONTROLLER
001012                  FILBUF:
001012  603600  004000  DTOCHK:  TLNE    IOS,DMPMOD       ;DUMP MODE?
001013  254000  001130'          JRST    DMPFIL           ;YES, FILL BUFFER FROM LIST
001014  205066  000010          MOVSI   TAC,@DEVOAD(DEVDAT) ;LOCATION OF BUFFER
001015  270040  002104'          ADD     TAC,[XWD 1,BUF]  ;SET TO STORE IN MONITOR BUFFER
001016  251040  002076'          BLT     TAC,BUF+177      ;GO BLT IT
001017  336000  001677'          SKIPN   BUF              ;GIVE UP TAPE IF
001020  254000  001445'          JRST    THRUTP           ;NO BUFFER TO OUTPUT
001021  621000  000200          TLZ     IOS,NOLINK       ;ASSUME A LINK IS GIVEN
```

```
555  001022  602000  000100  OUFULL: TRNE    IOS,UDSD        ;NON-STANDARD?
556  001023  254000  001043'         JRST    OUTBL2          ;YES, NO FILE-STRUCTURED OPERATIONS
557  001024  574000  001677'         HLRE    BLK,BUF         ;IS IT?
558  001025  321200  001050'         JUMPL   BLK,LSTBLK      ;YES, - LAST BLOCK OF FILE
559  001026  326200  001033'         JUMPN   BLK,OUTBLK      ;IF NON-0 - YES
560  001027  623000  020000         TLZE    IOS,DMPCLS      ;NO, LAST BLOCK OF A DUMP FILE?
561  001030  254000  001033'         JRST    OUTBLK          ;YES, LINK MUST STAY 0
562  001031  260140  000611' OUCOMP: PUSHJ   PDP,USRFRE      ;COMPUTE NEXT BLOCK
563  001032  661000  000200         TLO     IOS,NOLINK      ;THIS BLOCK NOT LINKED
564  001033  506200  001677' OUTBLK: HRLM    BLK,BUF         ;SAVE LINK IN 1ST WORD OF BLOCK
565  001034  200406  000013         MOVE    TEM,FSTBLK(DEVDAT) ;STORE 1ST BLOCK OF FILE IN WORD
566  001035  137400  002105'         DPB     TEM,[POINT 18,BUF,27]
567  001036  550406  000016         HRRZ    TEM,OBLK(DEVDAT) ;BLOCK TO WRITE NOW
568  001037  542206  000016         HRRM    BLK,OBLK(DEVDAT) ;BLOCK TO WRITE NEXT
569  001040  200200  000010         MOVE    BLK,TEM
570  001041  260140  000536'         PUSHJ   PDP,MARKDR      ;MARK BLOCK TAKEN IN DIRECTORY
571  001042  334200  000010         SKIPA   BLK,TEM
572  001043  550206  000016 OUTBL2: HRRZ    BLK,OBLK(DEVDAT)
573  001044  260140  000760'         PUSHJ   PDP,STOIOS
574
575
576  001045  260140  001250' WRTBLK: PUSHJ   PDP,FNOBLK      ;GO SEARCH FOR BLOCK
577  001046  200100  002106'         MOVE    TAC1,[RLKO DC,700] ;HERE WE ARE - GO WRITE
578  001047  254000  001202'         JRST    RDWRT
```

```
  579                                   ;WRITE LAST BLOCK
  580  001050  201200  000000  LSTBLK: MOVEI   BLK,0            ;LINK=0
  581  001051  254000  001033'         JRST    OUTBLK           ;GO WRITE LAST BLOCK
  582
  583                                   ;TRYING TO WRITE DIRECTORY - STORE IN CORE
  584  001052  201066  000010  COR2HM: MOVEI   TAC,@DEVDAD(DEVDAT)  ;WHERE IT IS
  585  001053  505041  000001         HRLI    TAC,1(TAC)
  586  001054  540046  000014         HRR     TAC,DLOC(DEVDAT)        ;WHERE TO PUT IT
  587  001055  201101  000177         MOVEI   TAC1,177(TAC)
  588  001056  251042  000000         BLT     TAC,(TAC1)
  589  001057  661000  003400         TLO     IOS,CHNGDR       ;REMEMBER TO WRITE IT OUT
  590  001060  205100  400000         MOVSI   TAC1,400000
  591  001061  436106  000024         IORM    TAC1,DEVMOD(DEVDAT) ;DIR. IS NOW IN@CORE
  592  001062  627000  004000  FAKADV: TLZN    IOS,DMPMOD
  593  001063  264140  000000         PUSHJ   PDP,ADVBFE       ;ADVANCE BUFFERS
  594  001064  255000  000000         JFCL
  595  001065  621000  000200         TLZ     IOS,NOLINK       ;DIRECTORY BLOCK IS NOT LINKED
  596  001066  402006  000016         SETZM   OBLK(DEVDAT)
  597  001067  254070  001044'         JRST    STOIOS
```

```
598                                     ;SET UP POINTERS AND STUFF FOR DUMP-MODE
599  001070  661000  004000  DMPSET: TLO     IOS,DMPMOD      ;NO, LIGHT BIT
600  001071  260140  000712'         PUSHJ   PDP,GETDT       ;GET CONTROL
601  001072  505600  000007         HRLI    UUO,PROG
602  001073  260140  000000         PUSHJ   PDP,COMCHK      ;CHECK VALIDITY OF LIST
603  001074  254000  001614'         JRST    SVADER          ;NG, GIVE ADRESS ERROR
604  001075  336060  000014         SKIPN   TAC,@UUO        ;OK, NULL LIST?
605  001076  254000  001104'         JRST    DMPTS1          ;YES, RETURN
606                                     IFN     CPBIT, <
607  001077  602000  000100         TRNE    IOS,UDSD        ;NON-STD DUMP MODE?
608  001100  364240  001106'         SOJA    DAT,TOUSER      ;YES, GO DO DIRECT I/O
609                                     >
610  001101  370000  000014  DMPST2: SOS     UUO             ;NO, SAVE START OF LIST (-1)
611  001102  202606  000020         MOVEM   UUO,DMPLST(DEVDAT)
612  001103  254000  000473'         JRST    CPOPJ1
613
614  001104  262140  000001  DMPTS1: POP     PDP,TAC
615  001105  254000  001457'         JRST    THRUTD
616
617
618                                     IFN CPBIT, <
619                                     ;HERE TO START DUMP-MODE INTO USER AREA DIRECTLY
620  001106  322700  001636'  TOUSER: JUMPE   AC2,NOBLK0      ;CANT READ BLK 0 IN NON-STD DUMP MODE
621  001107  240240  777771         ASH     DAT,-7          ;NUMBER OF WRDS IN LIST /200
622  001110  350000  000005         AOS     DAT
623  001111  202240  001672'         MOVEM   DAT,BLKCNT      ;SAVE TO UPDATE POSITION
624  001112  201620  000014         MOVEI   UUO,@UUO        ;REAL ADDRESS OF LIST
625  001113  202600  001673'         MOVEM   UUO,USPNTR      ;SAVE IT
626  001114  202600  001674'         MOVEM   UUO,SVPNTR
627  001115  271047  000000         ADDI    TAC,(PROG)      ;RELOCATE ADDRESS OF 1ST IOWD
628  001116  202040  001660'         MOVEM   TAC,PNTR        ;AND SAVE IT
629  001117  200040  002107'         MOVE    TAC,[JSR DMPADV] ;SET UP LOC FOR WHEN
630  001120  202040  000000         MOVEM   TAC,DCLOC1      ;IOWD IS EXHAUSTED
631  001121  552340  001675'         HRRZM   PROG,ADRPRG     ;SAVE JUST ADDRESS OF PROG
632  001122  661000  040000         TLO     IOS,NOBUF       ;INDICATE DIRECTLY TO USER
633  001123  256023  000001         XCT     @1(PDP)         ;TURN ON/OFF IO
634  001124  607000  000020         TLNN    IOS,IO
635  001125  254000  001103'         JRST    CPOPJ1          ;READING - CONTINUE
636  001126  262140  000001         POP     PDP,TAC         ;WRITING - THIS WILL SAVE LOTS OF TIME
637  001127  254000  001043'         JRST    OUTBL2
638                                     >
```

```
                                ;FILL OUTPUT BUFFER FROM LIST
001130  205100  777601  DMPFIL: MOVSI   TAC1,-177
                                IFE     CPBIT, <
                                TRNE    IOS,UDSD
                                SUB     TAC1,ONEONE     ;200 DATA WORDS IF NON-STANDARD
                                >
001131  260140  001157' DMPFLB: PUSHJ   PDP,NXTCOM       ;GET NEXT COMMAND
001132  254000  001140'         JRST    DMPOTH          ;END OF LIST
001133  200401  000000  DMPFLA: MOVE    TEM,(TAC)       ;GET NEXT WORD
001134  202402  001700'         MOVEM   TEM,BUF+1(TAC1) ;INTO BUFFER
001135  252100  001150'         AOBJP   TAC1,DMPOVR     ;BUFFER FULL IF GOES
001136  253040  001133'         AOBJN   TAC,.-3         ;GET NEXT WORD FROM COMMAND
001137  254000  001131'         JRST    DMPFLB          ;GET NEXT COMMAND

001140                  DMPOTH:
                                IFE     CPBIT, <
                                TRNN    IOS,UDSD
                                >
001140  552100  001677'         HRRZM   TAC1,BUF        ;LIST RAN OUT  SAVE WORD COUNT

001141  402002  001700'         SETZM   BUF+1(TAC1)     ;ZERO REST OF BUFFER
001142  551102  001701'         HRRZI   TAC1,BUF+2(TAC1)
001143  303100  002076'         CAILE   TAC1,BUF+177    ;JUST ZERO 1 WORD IF AT TOP
001144  254000  001022'         JRST    OUFULL
001145  505102  777777         HRLI    TAC1,-1(TAC1)
001146  251100  002076'         BLT     TAC1,BUF+177    ;****TEST IF TOP OF BUFFER
001147  254000  001022'         JRST    OUFULL          ;NOW WRITE BUFFER

                                ;BUFFER FULL BEFORE END OF COMMAND
001150  253040  001153' DMPOVR: AOBJN   TAC,.+3         ;WAS THAT LAST WORD OF COMMAND?
001151  260140  001157'         PUSHJ   PDP,NXTCOM      ;YES, GET NEXT
001152  201040  000000         MOVEI   TAC,0
001153  202046  000021         MOVEM   TAC,SVDWRD(DEVDAT) ;NO, SAVE REMAINDER OF COMMAND
001154  201040  000177  DMPOVA: MOVEI   TAC,177
                                IFE     CPBIT, <
                                TRNN    IOS,UDSD
                                >
001155  202040  001677'         MOVEM   TAC,BUF         ;WD CNT =177
001156  254000  001022'         JRST    OUFULL          ;GO WRITE PART OF STUFF


                                ;GET NEXT COMMAND FROM LIST
001157  336006  000020  NXTCOM: SKIPN   DMPLST(DEVDAT)  ;END OF COMMANDS?
001160  254000  001170'         JRST    NXTCM2          ;YES, RETURN
001161  354046  000020         AOSA    TAC,DMPLST(DEVDAT) ;GET NEXT COMMAND
001162  542046  000020  NXTCM1: HRRM    TAC,DMPLST(DEVDAT) ;STORE GO-TO ADDRESS
001163  200060  000001         MOVE    TAC,@TAC        ;GET COMMAND
001164  322040  001170'         JUMPE   TAC,NXTCM2      ;END OF LIST
001165  327040  001162'         JUMPG   TAC,NXTCM1      ;GO-TO WORD
001166  271047  000000         ADDI    TAC,(PROG)      ;REAL COMMAND - ADD RELOCATION
001167  344040  001125'         AOJA    TAC,CPOPJ1      ;AND RETURN


                                ;END OF DUMP-MODE LIST
001170  402006  000021  NXTCM2: SETZM   SVDWRD(DEVDAT)  ;ZERO POINTERS
```

```
  692  001171  402006  000020          SETZM   DMPLST(DEVDAT)
  693  001172  263140  000000          POPJ    PDP,
  694
  695
  696  001173  332046  000021  DMPFLC: SKIPE   TAC,SVDWRD(DEVDAT)      ;IS THERE ANOTHER COMMAND
  697  001174  254000  001133'         JRST    DMPFLA                  ;YES. GET IT
  698  001175  254000  001551'         JRST    DMPTH2                  ;NO. THROUGH
```

```
                    000210  DTC=210
                    000214  DTS=214

                            ;IO INTERFACE
001176  607000  004000  READBF: TLNN    IOS,DMPMOD      ;HAVE CONTROL IF DUMP-MODE
001177  260140  000712'        PUSHJ   PDP,GETDT       ;GET DT CONTROL

001200  260140  001250' READBC: PUSHJ   PDP,FNDBLK      ;SEARCH FOR RIGHT BLOCK
001201  200100  002110'        MOVE    TAC1,[BLKI DC,300] ;FOUND IT - START READING

                            ;HERE WITH BLK=BLOCK NUMBER, TAC1=FUNCTION.  START SEARCH
001202  261140  000001  RDWRT:  PUSH    PDP,TAC
001203  502100  001227'        HLLM    TAC1,IOWD       ;BLKI OR BLKO
001204  505040  777600        HRLI    TAC,-200
001205  607000  010000        TLNN    IOS,RWDIR       ;WRITING (READING) DIRECT?
001206  334040  001654'        SKIPA   TAC,BFPNTR      ;NO. INTO BUF
001207  540046  000014        HRR     TAC,DLOC(DEVDAT)        ;YES. LOC OF DIRECTORY
001210  370000  000001        SOS     TAC             ;NO, SET FOR FORWARD READ
001211                    IOGO:   IFN     CPBIT, <

001211  607000  040000        TLNN    IOS,NOBUF       ;POINTER ALREADY SET UP IF DIRECT IO
                            >
001212  202040  001660'        MOVEM   TAC,PNTR        ;POINTER
001213  434100  001667'        OR      TAC1,UNIT       ;UNIT AND PI CHAN
001214  202100  001670'        MOVEM   TAC1,COMAND     ;SAVE COMMAND FOR READ OR WRITE
001215  262140  000001        POP     PDP,TAC         ;RESTORE SEARCH COMMAND
001216  720200  000000        CONO    DC,DCIN         ;SET DATA CONTROL TO READ
001217  700600  000000        CONO    PI,DCON         ;TURN ON DC PI CHANNEL
001220  505040  000037        HRLI    TAC,37
001221  700600  000000        CONO    PI,PIOFF
001222  721201  000000        CONO    DTC,(TAC)       ;START TAPE MOVING
001223  546040  001357'        HLRM    TAC,DTCINT
001224  700600  000000        CONO    PI,PION
001225  263140  000000        POPJ    PDP,            ;DISMISS INTERRUPT


                            ;HERE FOR ANY DATA WORD WITH TAPE IN REVERSE
001226  000000  000000  RVERS:  0
001227  720000  001660' IOWD:   BLKI    DC,PNTR         ;READ (WRITE) A WORD
001230  254000  001234'        JRST    RVTHRU          ;POINTER RAN OUT
001231  370000  001660'        SOS     PNTR            ;POINTER HAS TO BACK UP
001232  370000  001660'        SOS     PNTR
001233  254520  001226'        JEN     @RVERS          ;DISMISS THE INTERRUPT
```

```
                                        ;HERE WHEN POINTER RUNS OUT IN REVERSE
001234  264000  001241' RVTHRU: JSP     SHUTDN          ;SHUT DOWN DATA CONTROL
001235  254520  001236'         JEN     @RVERS          ;DISMISS

                                        ;HERE WHEN POINTER RUNS OUT FORWARD
001236  000000  000000  DTATHR: 0
001237  264000  001241'         JSP     SHUTDN          ;SHUT DOWN DATA CONTROL
001240  254520  001236'         JEN     @DTATHR         ;DISMISS


001241  000000  000000  SHUTDN: 0
001242  720300  010000          CONSZ   DC,10000        ;DATA MISSED?
001243  476000  001671'         SETOM   ERRFLG          ;YES, SET SWITCH
001244  721340  000400          CONSO   DTC,400         ;READING?
001245  720200  000000          CONO    DC,0            ;YES, TURN  OFF DC
001246  700600  000000          CONO    PI,DCOFF        ;TURN OFF DC PI
001247  254020  001241'         JRST    @SHUTDN
```

```
                        ;COME HERE TO START READING BLOCK NUMBERS
                                EXTERN PION,PIOFF
                        
001250  552200  001663' FNDBLK: HRRZM   BLK,BLOCK        ;BLOCK WE'RE LOOKING FOR
001251  402000  001665'         SETZM   ERRCNT
001252  402000  001671' FNDBL2: SETZM   ERRFLG          ;RESET DATA MISSED FLAG
001253  135040  000000         LDB     TAC,PUNIT       ;GET UNIT NUMBER
001254  242040  000003         LSH     TAC,3           ;POSITION IT
001255  660040  000000         TRO     TAC,DTCCHN      ;ADD PI CHANNEL
001256  202040  001667'         MOVEM   TAC,UNIT        ;SAVE
001257  200400  002111'         MOVE    TEM,[JSR SRCH]  ;SET UP  INTERRUPT CELLS
001260  202400  000000         MOVEM   TEM,DCLOC
001261  200400  002112'         MOVE    TEM,[JSR DTATHR]
                                IFN     CPBIT, <
001262  607000  040000         TLNN    IOS,NOBUF       ;DCLOC1 ALREADY SET UP IF DIRECT IO
                        >
001263  202400  001120'         MOVEM   TEM,DCLOC1      ;FOR FORWARD DATA OPERATIONS
001264  721300  020000         CONSZ   DTC,20000       ;IS TAPE ALREADY MOVING?
001265  254000  001277'         JRST    FNDBL4          ;YES

001266  660040  323200         TRO     TAC,323200      ;NO. SET READ BLK NOS,, START DELAY
001267  607000  001000         TLNN    IOS,RVERSE      ;MOVE TAPE BACKWARDS?
001270  660040  010000         TRO     TAC,10000       ;YES
001271                  FNDBL3: IFN     CPBIT, <
001271  261140  000000         PUSH    PDP,IOS
001272  621000  737777         TLZ     IOS,NOBUFC      ;SET UP DIRCTN NON-0 IF DIRECT IO
001273  512000  001676'         HLLZM   IOS,DIRCTN
001274  262140  000000         POP     PDP,IOS
                        >
001275  202006  000002         MOVEM   IOS,DEVIOS(DEVDAT)      ;RESTORE IOS IN CORE
001276  263140  000000         POPJ    PDP,            ;AND EXIT

                        ;HERE TO INITIATE READ OF A TAPE THAT IS MOVING ALREADY
001277  721300  010000 FNDBL4: CONSZ   DTC,10000       ;GOING BACKWARDS?
001300  660040  010000         TRO     TAC,10000       ;YES
001301  660040  323200         TRO     TAC,323200      ;SET TO READ BLOCKS, NO DELAY
001302  254000  001271'         JRST    FNDBL3          ;GO TELL CONTROL
```

```
                                ;INTERRUPT HERE TO READ A BLOCK NUMBER
001303  000000  000000  SRCH:   0
001304  202040  001661'          MOVEM   TAC,TEMP          ;SAVE TAC
001305  720040  000001          DATAI   DC,TAC            ;READ A BLOCK NUMBER
001306  274040  001663'          SUB     TAC,BLOCK         ;;PRESENT BLOCK - TARGET BLOCK
                                IFN     CPBIT, <
001307  332000  001676'          SKIPE   DIRCTN            ;IF DIRECT IO
001310  721340  010000          CONSO   DTC,10000         ;MUST READ DATA FORWARD
                                >
001311  322040  001340'          JUMPE   TAC,FOUND         ;THERE IF ZERO
001312  540040  001667'          HRR     TAC,UNIT          ;NO. SET UP READ OF NEXT BLOCK
001313  721300  010000          CONSZ   DTC,10000         ;GOING IN REVERSE
001314  650040  002113'          TDC     TAC,[XWD 400000,100000] ;YES, SWITCH TURN-AROUND TEST
001315  331000  000001          SKIPL   TAC               ;TURN AROUND?
001316  640040  012000          TRC     TAC,12000         ;YES, SWITCH DIR., TURN AROUND DELAY
001317  721201  320200          CONO    DTC,320200(TAC);TELL CONTROL
001320  720230  001216'          CONO    DC,DCIN           ;TELL DATA CONTROL ALSO
001321  200040  001661' SRCHXT: MOVE    TAC,TEMP          ;RESET TAAC
001322  254520  001303'          JEN     @SRCH             ;AND EXIT THE INTERRUPT


                                IFN CPBIT, <
                                ;HERE WHEN DUMP-MODE POINTER RUNS OUT
001323  000000  000000  DMPADV: 0
001324  202040  001661'          MOVEM   TAC,TEMP
001325  354040  001673'          AOSA    TAC,USPNTR        ;ADVANCE LOC OF POINTER
001326  542040  001673' DMPAV1: HRRM    TAC,USPNTR
001327  336060  000001          SKIPN   TAC,@TAC          ;END OF LIST?
001330  254000  001336'          JRST    DMPAV3            ;YES, STOP TAPE
001331  270040  001675'          ADD     TAC,ADRPRG        ;ADD RELOCATION
001332  327040  001326'          JUMPG   TAC,DMPAV1
001333  202040  001660'          MOVEM   TAC,PNTR          ;NEW POINTER
001334  200040  001661' DMPAV2: MOVE    TAC,TEMP          ;RESTORE TAC
001335  254520  001323'          JEN     @DMPADV


001336  264000  001241' DMPAV3: JSR     SHUTDN            ;SHUT DOWN TAPE
001337  254000  001334'          JRST    DMPAV2            ;RESTORE TAC AND EXIT THE INTERRUPT

                                >
```

```
835                                     ;HERE WHEN CORRECT BLOCK NUMBER IS FOUND
836  001340  721300  012000  FOUND:  CONSZ   DTC,10000          ;GOING FORWARD?
837  001341  254000  001351'          JRST    IORVRS            ;NO
838  001342  200040  001227'          MOVE    TAC,IOWD          ;YES, SET BLKI/BLKO INTO PI LOC
839  001343  202040  001260' END1:    MOVEM   TAC,DCLOC
840  001344  200040  001670'          MOVE    TAC,COMAND        ;PICK UP READ OR WRITE COMMAND
841  001345  602040  000400          TRNE    TAC,400           ;WRITE?
842  001346  720200  000000          CONO    DC,DCOUT          ;YES, CONDITION DATA CONTROL
843  001347  721231  360000          CONO    DTC,360000(TAC)   ;START DATA FLOW
844  001350  254000  001321'          JRST    SRCHXT            ;AND LEAVE
845
846  001351  201040  010000  IORVRS: MOVEI   TAC,10000         ;SET IO FOR REVERSE
847  001352  436040  001670'          ORM     TAC,COMAND
848  001353  201040  000177          MOVEI   TAC,177
849  001354  272040  001660'          ADDM    TAC,PNTR          ;READ FROM TOP OF BUFFER DOWN
850  001355  200040  002114'          MOVE    TAC,[JSR RVERS]   ;HAVE TO DO SOME STUFF AT INTERRUPT
851  001356  254000  001343'          JRST    END1
```

```
852                                     ;INTERRUPT HERE FOR FLAG CHANNEL
853  001357  721740  000037  DTCINT: CONSO   DTS,37          ;INTERRUPT FOR DECTAPE?
854  001360  254000  001360'          JRST    .              ;NO. GO AWAY
855  001361  721340  040000          CONSO   DTC,40000      ;YES. JOB DONE ENABLED?
856  001362  254000  001522'          JRST    BLKNUM         ;NO. READING BLOCK NUMBERS
857  001363  721300  004000          CONSZ   DTC,4000       ;YES. TIME FLAG ENABLED?
858  001364  254000  001510'          JRST    TIMINT         ;YES. CHECK IF THIS IS A TIME INTERRUPT
859  001365  720200  000000  DTCIN1: CONO    DC,0           ;NO. TURN OFF DATA CONTROL
860  001366  264000  000000          JSR     DTCSAV         ;SAVE ACS
861  001367  200300  001656'          MOVE    DEVDAT,USEWRD  ;RESTORE DEVDAT
862  001370  200006  000002          MOVE    IOS,DEVIOS(DEVDAT) ;AND IOS
863  001371  200340  001657'          MOVE    PROG,USEPRG
864  001372  721700  000001          CONSZ   DTS,1          ;JOB DONE LIT?
865  001373  721700  000116          CONSZ   DTS,116        ;AND NO ERRORS?
866  001374  254000  001616'          JRST    ERRS           ;NO. ERROR
867  001375  332000  001671'          SKIPE   ERRFLG         ;DATA MISSED?
868  001376  254000  001616'          JRST    ERRS           ;YES. TOO BAD
869
870                                     ;DATA WAS READ IN OR WRITTEN OUT FINE
871  001377  603000  004000  DTCIN2: TLNE    IOS,DMPMOD     ;DUMP MODE?
872  001400  254000  001567'          JRST    DMPTHR         ;YES. GO ELSEWHERE
873  001401  607000  002000          TLNN    IOS,SINGL
874  001402  254000  001405'          JRST    .+3
875  001403  135040  002105'          LDB     TAC,[POINT 18,BUF,27]  ;GET 1ST BLOCK NO. IF READ
876                                                         ;CAME FROM LOOKUP
877  001404  202046  000015          MOVEM   TAC,IBLK(DEVDAT)  ;STORE IN DDB
878  001405  623000  000001          TLZE    IOS,IOW        ;NO. IN IO WAIT?
879  001406  260140  000000          PUSHJ   PDP,SETIOD     ;YES. TAKE OUT OF WAIT
880  001407  623000  012000          TLZE    IOS,SINGL+RWDIR ;DIRECTORY OPERATION?
881  001410  254000  001445'          JRST    THRUTP         ;YES. LEAVE
882  001411  603000  000020          TLNE    IOS,IO         ;WRITING?
883  001412  254000  001467'          JRST    OUTHRU         ;YES
```

```
                        EXTERN  JBTSTS,PJOBN


                        ;HERE ON END OF AN INPUT BLOCK
001413  561100  000177          HRROI   TAC1,177         ;MASK OUT 1ST-BLK DATA
001414  606000  000100          TRNN    IOS,UDSD         ;UNLESS IN NON-STD
001415  406100  001677'         ANDM    TAC1,BUF
001416  201066  000007          MOVEI   TAC,@DEVIAD(DEVDAT) ;WHERE TO STORE BLOCK
001417  270040  002115'         ADD     TAC,[XWD BUF,1];FROM BUF TO THERE
001420  201101  000177          MOVEI   TAC1,177(TAC)
001421  251042  000000          BLT     TAC,(TAC1)       ;TRANSFER IT
001422  554200  001677'         HLRZ    BLK,BUF          ;NEXT BLOCK TO READ
001423  602000  000100          TRNE    IOS,UDSD
001424  354206  000015          AOSA    BLK,IBLK(DEVDAT) ;CONTIGUOUS BLOCKS IF NON-STD
001425  542206  000015          HRRM    BLK,IBLK(DEVDAT) ;SAVE IN DDB
001426  602000  700000          TRNE    IOS,IODTER+IODERR+IOIMPM ;WAS THERE AN INPUT ERROR?
001427  254000  001440'         JRST    THRUIN           ;YES. DONT ADVANCE BUFFERS
001430  260140  000747'         PUSHJ   PDP,ADVBFF       ;GET NEXT BUFFER
001431  254000  001440'         JRST    THRUIN           ;EMPTY BUF NOT AVAILABLE

001432  332000  000004          SKIPE   BLK              ;EXIT  IF EOF OR BLOCK TOO LARGE
001433  303200  001101          CAILE   BLK,TOPBLK       ;THE ERROR WILL BE CAUGHT ON THE
001434  254000  001440'         JRST    THRUIN           ;UUO LEVEL NEXT TIME AROUND
001435  375000  001664'         SOSGE   QUANTM           ;HAS TAPE HAD CHAN LONG ENOUGH?
001436  337000  000713'         SKIPG   DTREQ            ;YES. ANYONE ELSE WANT IT?
001437  254000  001200'         JRST    READBC           ;NO. READ NEXT BLOCK


001440  550046  000016  THRUIN: HRRZ    TAC,OBLK(DEVDAT) ;TAPE ALSO BEING WRITTEN?
001441  326040  001445'         JUMPN   TAC,THRUTP       ;YES. DONT CHANGE REVERSE BIT
001442  621000  001000          TLZ     IOS,RVERSE       ;NO. SET IOS BIT TO CORRECT DIRECTION
001443  721300  010000          CONSZ   DTC,10000
001444  661000  001000          TLO     IOS,RVERSE
```

```
916                                     ;HERE WHEN TAPE IS DONE
917  001445  721340  020000  THRUTP: CONSO   DTC,20000          ;ON INTERRUPT LEVEL?
918  001446  254000  001457'          JRST    THRUTD            ;NO. TAPE IS NOT MOVING
919  001447  200040  001667'          MOVE    TAC,UNIT          ;SET TO STOP TAPE
920  001450  721300  010000           CONSZ   DTC,10000         ;GOING REVERSE?
921  001451  660040  010000           TRO     TAC,10000         ;YES
922  001452  721201  245000           CONO    DTC,245000(TAC)   ;STOP TAPE, WITH TIME FLAG INTERRUPT ON
923                                                             ;ENABLE JOB DONE AS A FLAG FOR DTCINT
924  001453  371000  000000  THRUTA: SOSL    DCREQ              ;GIVE UP DATA CONTROL (DECTAPE CONTROL
925  001454  476000  000000           SETOM   DCAVAL             ;WILL BE GIVEN UP AT NEXT INTERRUPT)
926  001455  621000  044000           TLZ     IOS,DMPMOD+NOBUF  ;RESET DUMP-MODE BIT
927  001456  254000  000000           JRST    CLRACT  ;RESET IOACT AND RETURN
928
929  001457  260140  001453' THRUTD: PUSHJ   PDP,THRUTA         ;GIVE UP DATA CONTROL
930  001460  371000  001436'          SOSL    DTREQ             ;GIVE UP DECTAPE CONTROL
931  001461  476000  000000           SETOM   DTAVAL
932  001462  720200  000000           CONO    DC,0              ;SHUT OFF DATA CONTROL
933  001463  700600  001246'          CONO    PI,DCOFF          ;AND TURN OFF ITS PI CHANNEL
934  001464  721200  000000           CONO    DTC,0             ;SHUT DOWN DATA CONTROL
935  001465  513000  001357'          HLLZS   DTCINT
936  001466  263140  000000           POPJ    PDP,              ;AND LEAVE
```

```
937                                     EXTERN  CLOCK
938
939                             ;HERE ON END OF OUTPUT BLOCK
940  001467  260140  001063' OUTHRU: PUSHJ   P,ADVBFE          ;GET NEXT BUFFER
941  001470  254000  001445'         JRST    THRUTP           ;NOT FULL
942  001471  322240  001445'         JUMPE   BLK,THRUTP       ;DONE IF BLK=0
943  001472  550046  000016          HRRZ    BLK,OBLK(DEVDAT) ;NEXT BLOCK TO WRITE
944  001473  303200  001101          CAILE   BLK,TOPBLK       ;LEGAL?
945  001474  254000  001445'         JRST    THRUTP           ;NO. CATCH ERROR ON UUO LEVEL
946  001475  375000  001664'         SOSGE   QUANTM           ;YES. HAD CHAN LONG ENOUGH?
947  001476  337000  001460'         SKIPG   DTREQ            ;AND SOMEONE ELSE WANT IT?
948  001477  254000  001012'         JRST    FILBUF           ;NO. GO WRITE NEXT BLOCK
949  001500  254000  001445'         JRST    THRUTP           ;YES. GIVE UP TAPE
950
951
952                             ;TURN TAPE AROUND AFTER END-ZONE INTERRUPT
953  001501  202040  001662' TURN:   MOVEM   TAC,TEMPA         ;SAVE TAC
954  001502  200040  001667'         MOVE    TAC,UNIT         ;UNIT AND CHANNEL
955  001503  721340  010000          CONSO   DTC,10000        ;IN REVERSE?

956  001504  660040  010000          TRO     TAC,10000        ;NO. NO WILL BE
957  001505  721201  322200          CONO    DTC,322200(TAC);READ BLOCK NOS IN OPPOSITE DIRECTION
958  001506  200040  001662'         MOVE    TAC,TEMPA        ;RESTORE TAC
959  001507  254520  000000          JEN     @DTCCHL          ;AND EXIT THE INTERRUPT
960
961  001510  721740  000020 TIMINT: CONSO   DTS,20           ;TIME FLAG INTERRUPT ON?
962  001511  254000  001527'         JRST    SPRIUS           ;NO. THIS IS A SPURIOUS INTERRUPT
963  001512  250040  001532'         EXCH    TAC,TIMREQ       ;ANOTHER TAPE CAN NOT BE
964  001513  700600  001221'         CONO    PI,PIOFF         ;SELECTED FOR 25 MSEC SO
965  001514  136040  000000          IDPB    TAC,CLOCK        ;PUT IN A CLOCK REQUEST
966  001515  700600  001224'         CONO    PI,PION ;TO WAIT FOR 3 TICKS
967  001516  250040  001532'         EXCH    TAC,TIMREQ       ;BEFORE ALLOWING NEXT USER ON
968  001517  721200  000000          CONO    DTC,0            ;SHUT DOWN DTC
969  001520  513000  001357'         HLLZS   DTCINT
970  001521  254520  001507'         JEN     @DTCCHL          ;AND LEAVE
971
972
973                             ;HERE WITH AN INTERRUPT WHILE SEARCHING FOR BLOCK NUMBERS
974  001522  721700  000002 BLKNUM: CONSZ   DTS,2            ;END ZONE INTERRUPT?
975  001523  254000  001501'         JRST    TURN             ;YES. TURN AROUND
976  001524  721740  000014          CONSO   DTS,14           ;NO. ILLEGAL OP OR PARITY ERROR?
977  001525  254000  001360'         JRST    DTCINT+1         ;NO. NOT REALLY A DECTAPE INTERRUPT
978  001526  254000  001365'         JRST    DTCIN1           ;YES. GO HANDLE THE ERROR
979
980                             ;HERE ON A SPURIOUS DECTAPE INTERRUPT
981  001527  721240  001662' SPRIUS: CONI    DTC,TEMPA        ;READ DTC STATUS BITS
982  001530  721220  001662'         CONO    DTC,@TEMPA       ;GIVE A CONO TO DTC TO CLEAR INTERRUPT
983  001531  254520  001521'         JEN     @DTCCHL          ;EXIT THE INTERRUPT
984
985  001532  001533' 000003 TIMREQ: XWD     .+1,3    ;E CLOCK TICKS
986  001533  371000  001476'         SOSL    DTREQ    ;ALLOW OTHER JOBS
```

```
 987  001534  476000  001461'          SETOM   DTAVAL  ;TO GET DT CONTROL NOW
 988  001535  263140  000000          POPJ    PDP,
```

```
  989                                   ;COME HERE ON END OF DUMP OR SAVE MODE BLOCK
  990  001536                           SVDMTH: IFN     CPBIT, <
  991  001536  603000  040000                   TLNE    IOS,NOBUF
  992  001537  254000  001561'                 JRST    USDMTH          ;HANDLE DIFFERENTLY IF DIRECT TO USER
  993                                   >
  994  001540  205100  777601                   MOVSI   TAC1,-177
  995                                           IFE     CPBIT, <
  996                                           TRNE    IOS,UDSD
  997                                           SUB     TAC1,ONEONE     ;SET UP TAC1 WITH COUNT
  998                                   >
  999  001541  607000  000020                   TLNN    IOS,IO
 1000  001542  254000  001556'                 JRST    SVDMIN          ;INPUT FILE
 1001  001543  550206  000016                   HRRZ    BLK,OBLK(DEVDAT) ;OUTPUT FILE, NEXT BLOCK
 1002  001544  322200  001550'                 JUMPE   BLK,DMPTHA      ;LAST BLOCK
 1003                                           IFE     CPBIT, <
 1004                                           TRNE    IOS,UDSD        ;IF NON-STD MODE
 1005                                           AOSA    OBLK(DEVDAT)    ;WRITE CONSECUTIVE BLOCKS
 1006                                   >
 1007  001545  307200  001101                   CAIG    BLK,TOPBLK      ;NOT LAST, LEGAL BLOCK NUMBER?

 1008  001546  263140  000000                   POPJ    PDP,            ;YES, RETURN
 1009
 1010  001547  660000  040000                   TRO     IOS,IOBKTL      ;BLOCK TOO LARGE
 1011  001550  262140  000001           DMPTHA: POP     PDP,TAC
 1012  001551  402006  000021           DMPTH2: SETZM   SVDWRD(DEVDAT)  ;ZERO DUMP-MODE STUFF
 1013  001552  402006  000020                   SETZM   DMPLST(DEVDAT)
 1014  001553  623000  000001           DMPTH3: TLZE    IOS,IOW         ;IN IO WAIT?
 1015  001554  260140  001406'                 PUSHJ   PDP,SETIOD      ;YES, RESTART JOB
 1016  001555  254000  001445'                 JRST    THRUTP
 1017
 1018                                   ;HERE ON END OF SAVE MODE INPUT BLOCK
 1019  001556  554200  001677'         SVDMIN: HLRZ    BLK,BUF         ;NEXT BLOCK NUMBER
 1020                                           IFE     CPBIT, <
 1021                                           TRNE    IOS,UDSD        ;NON-STANDARD?
 1022                                           AOSA    BLK,IBLK(DEVDAT) ;YES, READ CONSECUTIVE BLOCKS
 1023                                   >
 1024  001557  542206  000015                   HRRM    BLK,IBLK(DEVDAT) ;SAVE IN DDB
 1025  001560  254000  001167'                 JRST    CPOPJ1
 1026
 1027
 1028                                           IFN CPBIT, <
 1029                                   ;HERE WHEN THROUGH DUMP-MODE DIRECTLY TO USER
 1030  001561  201106  000015         USDMTH: MOVEI   TAC1,IBLK(DEVDAT)
 1031  001562  603000  000020                   TLNE    IOS,IO
 1032  001563  201106  000016                   MOVEI   TAC1,OBLK(DEVDAT)       ;SET TAC1 TO RIGHT BLOCK NUMBER
 1033  001564  200040  001672'                 MOVE    TAC,BLKCNT      ;UPDATE BLOCK COUNTER
 1034  001565  272042  000000                   ADDM    TAC,(TAC1)
 1035  001566  254000  001550'                 JRST    DMPTHA          ;THROUGH
 1036                                   >
```

```
1037                                    ;HERE WHEN THROUGH DUMP MODE BLOCK
1038  001567  260140  001536' DMPTHR: PUSHJ   PDP,SVDMTH       ;END OF BLOCK HOUSEKEEPING
1039  001570  254000  001173'         JRST    DMPFLC          ;FILL BUFFER FOR NEXT OUTPUT
1040
1041                                    ;HERE WHEN THROUGH READING A DUMP-MODE BLOCK
1042  001571  332046  000021  DMIFIL: SKIPE   TAC,SVDWRD(DEVDAT) ;PARTIAL COMMAND?
1043  001572  254000  001575'         JRST    DMIFLB          ;YES, CONTINUE
1044  001573  260140  001157' DMIFLA: PUSHJ   PDP,NXTCOM       ;NO, GET NEXT COMMAND
1045  001574  254000  001551'         JRST    DMPTH2          ;END OF LIST - THROUGH
1046  001575  200402  001700' DMIFLB: MOVE    TEM,BUF+1(TAC1) ;NEXT DATA WORD
1047  001576  202401  000000         MOVEM   TEM,(TAC)       ;GIVE TO USER
1048  001577  252100  001602'         AOBJP   TAC1,.+3        ;IF BUFFER IS FULL
1049  001600  253040  001575'         AOBJN   TAC,DMIFLB      ;GET NEXT WORD
1050  001601  254000  001573'         JRST    DMIFLA          ;GET NEXT COMMAND
1051
1052  001602  253040  001605'         AOBJN   TAC,.+3         ;BUFFER ID FULL, IS COUNT EXACTLY 177?
1053  001603  260140  001157'         PUSHJ   PDP,NXTCOM      ;THAT COM, IS DONE, GET NEXT
1054  001604  254000  001551'         JRST    DMPTH2          ;END OF LIST - THROUGH
1055  001605  202046  000021         MOVEM   TAC,SVDWRD(DEVDAT)  ;SAVE PARTIAL COMMAND FOR NEXT TIME
1056  001606  322200  001612'         JUMPE   BLK,DMPEOF      ;IF EOF - LIGHT BIT
1057  001607  307200  001101  RDNXT:  CAIG    BLK,TOPBLK      ;BLOCK LEGAL?
1058  001610  254000  001200'         JRST    READBC          ;GO READ BLOCK NUMBER
1059  001611  664000  040000         TROA    IOS,IOBKTL      ;LIGHT ERROR BIT
1060
1061
1062                                    ;EOF BEFORE ALL DATA IS IN - DUMP MODE
1063  001612  660000  020000  DMPEOF: TRO     IOS,IODEND      ;LIGHT EOF BIT
1064  001613  254000  001551'         JRST    DMPTH2          ;GIVE UP TAPE
```

```
1065  001614  260140  001551' SVADER: PUSHJ   PDP,DMPTH2        ;GIVE UP CONTROL
1066  001615  254000  000000          JRST    ADRERR           ;TYPE ERROR MESSAGE
1067
1068
1069                                ;COME HERE ON ERROR
1070  001616  350040  001665' ERRS:   AOS     TAC,ERRCNT       ;BUMP COUNT
1071                                  IFN     CPBIT, <
1072  001617  200100  001674'        MOVE    TAC1,SVPNTR
1073  001620  202100  001673'        MOVEM   TAC1,USPNTR
1074  001621  200102  000000        MOVE    TAC1,(TAC1)
1075  001622  270100  001675'        ADD     TAC1,ADRPRG
1076  001623  202100  001660'        MOVEM   TAC1,PNTR        ;RESET STUFF FOR POSSIBLE DIRECT IO
1077                                >
1078  001624  721740  000004        CONSO   DTS,4            ;IF ILLEGAL OP - DONT RETRY
1079  001625  303040  000000        CAILE   TAC,DTTRY        ;ENOUGH REREADS?
1080  001626  334000  000000        SKIPA                    ;YES. PERMANENT ERROR
1081  001627  254000  001252'        JRST    FNOBL2           ;NO. TRY AGAIN
1082
1083                                ;PERMANENT ERROR

1084  001630  721700  000010  PERMER: CONSZ   DTS,10
1085  001631  660000  100000          TRO     IOS,IODTER       ;PARITY
1086  001632  721740  000100          CONSO   DTS,100
1087  001633  332000  001671'        SKIPE   ERRFLG
1088  001634  660000  200000        TRO     IOS,IODERR       ;MISSED DATA
1089  001635  721700  000004        CONSZ   DTS,4
1090  001636  660000  400000  NOBLK0: TRO     IOS,IOIMPM       ;ILLEGAL OP
1091  001637  603000  004000        TLNE    IOS,DMPMOD       ;IF IN DUMP-MODE
1092  001640  254000  001567'        JRST    DMPTHR           ;CONTINUE WITH REST OF LIST
1093  001641  607000  012020        TLNN    IOS,IO+RWDIR+SINGL ;READING DATA?
1094  001642  254000  001377'        JRST    DTCIN2           ;YES. GIVE BUFFER TO USER
1095  001643  623000  000001        TLZE    IOS,IOW
1096  001644  260140  001554'        PUSHJ   PDP,SETIOD       ;OUT OF IO WAIT
1097  001645  627000  012000        TLZN    IOS,RWDIR+SINGL  ;DIRECTORY OPERATION?
1098  001646  254000  001445'        JRST    THRUTP           ;NO. RETURN TO USER
1099  001647  260140  001445'        PUSHJ   PDP,THRUTP       ;YES. STOP TAPE
1100  001650  205040  400000        MOVSI   TAC,400000       ;SET DIR NOT IN CORE BIT
1101  001651  412046  000004        ANDCAM  TAC,DEVMOD(DEVDAT)
1102  001652  135200  000000  BDDIR:  LDB     BLK,PJOBN        ;TELL ERRCON THE NUMER OF THE JOB
1103  001653  254000  000000          JRST    BADDIR           ;GO PRINT ERROR MESSAGE
```

```
1124  001654  777600  001677' RFPNTR: IOWD    200,BUF+1
1125  001655  000001  000001  ONEONE: XWD     1,1
1126  001656  000000  000000  USEWRD: 0
1127  001657  000000  000000  USEPRG: 0
1128  001660  000000  000000  PNTR:   0
1129  001661  000000  000000  TEMP:   0
1130  001662  200000  000000  TEMPA:  0
1131  001663  000000  000000  BLOCK:  0
1132  001664  000000  000000  QUANTM: 0
1133  001665  000000  000000  ERRCNT: 0
1134  001666  000000  000000  FNDTMP: 0
1135  001667  000000  000000  UNIT:   0
1136  001670  000000  000000  COMAND: 0
1137  001671  000000  000000  ERRFLG: 0
1138                          IFN     CPBIT, <
1139  001672  000000  000000  BLKCNT: 0
1140  001673  000000  000000  USPNTR: 0
1141  001674  000000  000000  SVPNTR: 0
1142  001675  000000  000000  ADRPRG: 0
1143  001676  000000  000000  DIRCTN: 0
1144                          >
1145  001677                  BUF:    BLOCK   200
1146  002077                  DTAEND: END
1147  002077  777777  777671
1148  002100  777752  000123
1149  002101  050000  000000
1150  002102  000026  000026
1151  002103  000040  020000
1152  002104  000001  001677'
1153  002105  101200  001677'
1154  002106  720100  000700
1155  002107  264000  001323'
1156  002110  720000  000300
1157  002111  264000  001333'
1158  002112  264000  001236'
1159  002113  400020  010000
1160  002114  264000  001226'
1161  002115  001677' 000001
```

NO ERRORS DETECTED

PROGRAM BREAK IS 002116

SYMBOL TABLE

```
AC1             000015  INT     AC2             000016  INT     AC3             000017  INT
ADRERR          001615' EXT     ADRPRG          001675'         ADVBFE          001467' EXT
ADVBFF          001430' EXT     BADDIR          001653' EXT     BDDIR           001652'
BFPNTR          001654'         BLK             000004          BLKCHK          000752'
BLKCNT          001672'         BLKNUM          001522'         BLKSRA          000420'
BLKSRB          000423'         BLKSRC          000415'         BLOCK           001663'
BUF             001677'         CALNXT          000624'         CHNGDR          000400
CLOCK           001514' EXT     CLRACT          001456' EXT     CLSDMP          000721'
COMAND          001670'         COMCHK          001073' EXT     CUR2HM          001052'
CPRIT   777777  777777          CPOPJ           000547' EXT     CPOPJ1          001560' EXT
DAT             000005  INT     DCAVAL          001454' EXT     DCIN            001320' EXT
DCLOC           001343' EXT     DCLOC1          001263' EXT     DCOFF           001463' EXT
DCON            001217' EXT     DCOUT           001346' EXT     DCREQ           001453' EXT
DECPTR          000455'         DEVCAT          000006  INT     DEVEXT          000012  INT
DEVFIL          000011  INT     DEVIAD          000007  INT     DEVIOS          000002  INT
DEVMOD          000004  INT     DEVOAD          000010  INT     DIRBLK          000144
DIRCHK          000471'         DIRCTN          001676'         DISPAD          000017
DLETF           000464'         DLOC            000014          DMIFIL          001571'
DMIFLA          001573'         DMIFLB          001575'         DMPADV          001323'

DMPAV1          001326'         DMPAV2          001334'         DMPAV3          001336'
DMPCLS          020000          DMPEOF          001612'         DMPFIL          001130'
DMPFLA          001133'         DMPFLB          001131'         DMPFLC          001173'
DMPI            000725'         DMPLST          000020          DMPMOD          004000
DMPO            000773'         DMPOTH          001140'         DMPOVA          001154'
DMPOVR          001150'         DMPSET          001070'         DMPST2          001101'
DMPTH2          001551'         DMPTH3          001553'         DMPTHA          001550'
DMPTHR          001567'         DMPTS1          001104'         DRPTR           000543'
DSER1           000360'         DSERCH          000356'         DTAEND          002077'
DTATHR          001236'         DTAVAL          001534' EXT     DTC             000210
DTCCHL          001531' EXT     DTCCHN          001255' EXT     DTCDDB          000000' INT
DTCDDS          000222  INT     DTCDIR          000022'         DTCDSP          000224'
DTCIN1          001365'         DTCIN2          001377'         DTCINI          000243' INT
DTCINT          001357' INT     DTCSAV          001366' EXT     DTCSRN          000000' INT
DTCCHK          001012'         DTREQ           001533' EXT     DTS             000214
DTTRY           001625' EXT     DVDIRI          400000  INT     ENTR            000472'
ENTR2           000476'         ENTRA           000506'         ENTRC           000526'
ENTRD           000570'         EOF             000756'         ERRCNT          001665'
ERRFLG          001671'         ERRS            001616'         FAKADV          001062'
FILBUF          001012'         FND1            001343'         FNDBL2          001252'
FNDBL3          001271'         FNDBL4          001277'         FNDBLK          001250'
FNDTMP          001646'         FOUND           001340'         FREXIT          000440'
FSTBLK          000013          FTCCL   777777  777777          FTDISK  777777  777777
FTLOGI  777777  777777          FTRC10  777777  777777          FTSWAP  777777  777777
GETDCD          000712' EXT     GETJT           000712'         GETF            000602'
HUNGST          000001  INT     IBLK            000015          IO              000020  INT
IOPKTL          040000  INT     IODEND          020000  INT     IODERR          200000  INT
IODTER          100000  INT     IOEND           000040  INT     IOGO            001211'
IOIMPM          400000  INT     IORVRS          001351'         IOS             000000  INT
IOW             000001  INT     IOWD            001227'         JBTADR          000000  EXT
JBTSTS          000040  EXT     JOBDDT          000762' EXT     JOBSAV          000000  EXT
LOOK            000247'         LOOKA           000256'         LOOKC           000307'
LOOKD           000331'         LSTBLK          001050'         LSTFRE          000442'
MARKDR          000536'         NAMSTR          000123          NEWENT          000553'
```

SYMBOL TABLE

```
NMFOUN  000372'        NMLOOK  000363'        NOBLK0  001636'
NOPLKS  000640'        NOBUF   040000         NOBUFC  737777
NOLINK  000200         NXTCM1  001162'        NXTCM2  001170'
NXTCOM  001157'        NXTERE  000433'        OBLK    000016
ONEONE  001655'        OUCOMP  001031'        OUFULL  001022'
OUT     000676' EXT    OUTBL2  001043'        OUTBLK  001033'
OUTHRU  001467'        PDP     000003  INT    PERMER  001630'
PIOFF   001513' EXT    PIOMOD  000666' EXT    PION    001515' EXT
PJOBN   001652' EXT    PNTR    001660'        PROG    000007  INT
PUNIT   001253' EXT    QUANT   000005         QUANTM  001664'
RDNXT   001647'        RDWRT   001222'        READBC  001200'
READRF  001176'        RELEA9  000000  EXT    RENAM   000315'
RENAM1  000332'        RENAM2  000335'        RENER1  000350'
RENER2  000352'        RVERS   001226'        RVERSE  001000
RVTHRU  001234'        RWDIR   010000         SD      000015  INT
SETACT  000720' EXT    SETI    000574'        SETIOD  001644' EXT
SETO    000576'        SETPTR  000426'        SETWD4  000560'
SHUTDN  001241'        SINGL   002000         SPACE   000006
SPRIUS  001527'        SRCH    001333'        SRCHXT  001321'
STOIOS  001047' EXT    STOTAC  000610' EXT    STOWD4  000271'
SVADER  001614'        SVDMIN  001556'        SVDMTH  001536'
SVDWRD  000021         SVPNTR  001674'        TAC     000001  INT
TAC1    000002  INT    TEM     000010  INT    TEMP    001661'
TEMPA   001662'        THRUIN  001440'        THRUTA  001453'
THRUTD  001457'        THRUTP  001445'        THSDAT  000502' EXT
TIMINT  001510'        TIMREQ  001532'        TOPBLK  001101
TOUSER  001176'        TPOPJ   000364' EXT    TPOPJ1  000000  EXT
TURN    001571'        UADCK1  000357' EXT    UCLS    000663'
UDSD    000100         UIN     000730'        UNIT    001667'
UOUT    000777'        UOUT2   001010'        UREL    000677'
USDMTH  001561'        USERPRG 001657'        USEWRD  001656'
USLSTA  000636'        USPNTR  001673'        USRDDT  000771' EXT
USRERA  000616'        USRERE  000611'        USRLST  000633'
UTPCLR  000642'        UUO     000014  INT    VDTASR  000012  INT
WAIT1   000711' EXT    WRTBLK  001045'        ZERCOR  000762'
```

```
DTCSRN      16      17#
DTCCHK     547#
DTREQ       20     460     907     930     947     986
DTS        700#    853     864     865     961     974     976    1078    1084    1086    1089
DTTRY       22    1079
DVAVAL      6#       6
DVCOP       6#       6
DVDIR       6#       6
DVDIRI      6#       6     220
DVDIS       6#       6
DVDSK       6#       6
DVDTA       6#       6
DVIN        6#       6
DVLNG       6#       6
DVLPT       6#       6
DVMTA       6#       6
DVOUT       6#       6
DVPTP       6#       6
DVPTR       6#       6
DVTTY       6#       6
ENTR        88     283#
ENTR2      287#    342
ENTRA      296#    355
ENTRB       6#       6
ENTRC      314#    360
ENTRD      307     357#
EOF        488     510#
ERRCNT     762    1070    1113#
ERRFLG     753     763     867    1087    1117#
ERRS       866     868    1070#
FAKADV     542     592#
FBMERR      6#       6
FILBUF     546#    948
FND1       839#    851
FNDBL2     763#   1081
FNDBL3     780#    794
FNDBL4     776     791#
FNDBLK     576     706     761#
FNDTMP    1114#
FNFERR      6#       6
FOUND      804     836#
FREXIT     249#    261     263
FRGSEG      6#       6
FSTBLK      38#    117     145     146     314     445     565
FT2REL      6#
FTATTA      6#
FTCCL      10#
FTCHEC      6#
FTDISK     10#
FTEXAM      6#
FTFINI      6#
FTGETT      6#
FTHALT      6#
```

	862	871	873	878	880	882	889	896	899	913	915	926	991	999
	1010	1014	1031	1059	1063	1085	1088	1090	1091	1093	1095	1097		
IOTEND	6#	6												
IOUSE	6#	6												
IOW	6#	6	878	1014	1095									
IOWC	6#	6												
IOWD	711	736#	838											
IPPERR	6#	6												
ITEM	6#	6												
JACCT	6#	6												
JBFADR	6#	6												
JBFCTR	6#	6												
JBFPTR	6#	6												
JBTADR	23													
JBTSTS	884													
JBUF	6#	6												
JDAT	6#	6												
JERR	6#	6												
JLOG	6#	6												
JNA	6#	6												
JOBDDT	473	517												
JOBSAV	176													
JRC	6#	6												
JWPOS	6#	6												
JWSIZ	6#	6												
JXPN	6#	6												
LISTSN	6													
LOOK	89	178#												
LOOKA	115#													
LOOKB	6#	6												
LOOKC	145#													
LOOKD	137#													
LSTBLK	558	580#												
LSTFRE	135	253#	404											
MARKOR	323#	570												
MEDDLE	6#	6												
NAMSTR	32#	193	427											
NECERR	6#	6												
NEWENT	286	340#												
NLEERR	6#	6												
NMFOUN	199	202#												
NMLOOK	194#	207												
NOBLKØ	620	1090#												
NORLKS	397	409#												
NORUF	57#	632	718	772	926	991								
NORUFC	58#	782												
NOLINK	50#	320	430	554	563	595								
NSCERR	6#	6												
NSFERR	6#	6												
NSHF	6#	6												
NSRBIT	6#	6												
NSWP	6#	6												
NXM	6#													

CODES	6#		
DISABL	6#		
ENABLE	6#		
NOSCHE	6#		
NOSHUF	6#		
QUEUES	6#		
SCHEDU	6#		
SHUFFL	6#		
STARTD	6#		
XP	6#	6	13