```
                        ;***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
                        IFE FTDISK+FTRC10+2,<
                        TITLE   ONCER - ONCE ONLY OPERATOR DIALOGUE AND IO DEVICE INIT.(BURROUGHS DISK)
                        >
                        IFE FTDISK+FTRC10+1,<
                        TITLE   ONCED - ONCE ONLY OPERATOR DIALOGUE AND IO DEVICE INIT.(DATA PRODUCTS DISK)
                        >
                        IFE FTDISK+FTRC10+0,<
                        TITLE   ONCEN - ONCE ONLY OPERATOR DIALOGUE AND IO DEVICE INIT.(NO DISK)
                        >
                        SUBTTL  T. HASTINGS/RCC/CHW  TS  04 JUN 69  V414
                        XP VONCE,414+
                                        ;PUT VERSION NUMBER IN GLOB LISTING AND LOADER STORAGE MAP

                        INTERNAL        FTRC10  ;THIS SINGLE SOURCE FILE MAY BE ASSEMBLED FOR EITHER
                                                ; THE NEW PDP-10 (BURROUGHS) DISK, OR THE OLD PDP-6 (DATA
                                                ; PRODUCTS) DISK (FTRC10 = -1 OR 0, RESPECTIVELY, IN S).

                        ;"ONCE" SHOULD BE THE LAST LOADED PROGRAM BEFORE SYSMAK AND DDT
                        ;THUS IF IT OVERFLOWS INTO THE USER AREA NO HARM IS DONE.

                        ;ONCE ONLY CODE IS DIVIDED INTO 3 PARTS
                        ; 1. MANDATORY ONCE ONLY(LINKSR) - NO CTY NEEDED
                        ;       (GOOD FOR REPEATED STARTUPS DURING DEBUGGING
                        ; 2. USUAL SHROT ONCE ONLY CODE(ONCE) - CTY FOR DATE AND TIME
                        ; 3. OPTIONAL ONCE ONLY CODE - CTY DIALOG TO CHANGE MONITOR
                        ;       OR REFRESH DISK(ALT-MODE INSTEAD OF CR AFTER TIME)
```

```
                                ;INITIALIZE PRIORITY CHANNELS AND SETUP INTERRUPT SERVICE ROUTINE CHAIN


                        000001  T=TAC                   ;DEFINE ACCUMULATOR T

                                ;LINK TO SERVICE ROUTINES

                                INTERNAL LINKSR,FT2REL
                                EXTERNAL JOBSYM,JOBPFI,XJBPFI

000000  200000  000000  LINKSR: 0                       ;CALLED WITH JSR FROM SYSINI
000001  205040  000000          MOVSI T,JOBPFI          ;SET LH OF XJBPFI WITH JOBPFI
000002  202040  000000          MOVEM T,XJBPFI          ;LOADER DORES NOT HANDLE GLOBALS IN LH
                                IFN FT2REL,<
000003  264000  000111'          JSR TSTREG              ;GO TEST IF TWO RELOC REGISTERS EXIST
                                >

                                                        ;HIGHEST LOC. IN JOB AREA PROTECTED FROM IO
```

```
                        ;MOVE SYMBOL TABLE UP IN MEMORY
                        ;EXEC MUST BE LOADED IN 15K OR LESS IF GOING INTO 16K MACHINE
                        ;FIRST FIND THE FIRST LOCATION OF NON-EXISTENT MEMORY

                                EXTERN DDTSYM

000004  403240  000000          SETZB   DAT,IOS         ;(ACCUMULATOR IOS MUST ALSO BE CLEARED FOR SCNSER)
000005  700200  010000          CONO APR,NXM            ;CLEAR NON-EX MEM FLAG
000006  201240  000400          MOVEI   TAC,+D256       ;IN CASE MEMORY HAS NO NON EXISTANT BANK$
000007  271240  002000          ADDI DAT,2000           ;TRY NEXT 1K BLOCK
000010  200125  000000          MOVE TAC1,(DAT)         ;REFERENCE THIS LOCATION
000011  700340  010000          CONSO APR,NXM           ;NON-EXISTENT?
000012  367040  000007'          SOJG TAC, .-3           ;NO,KEEP LOOKING AS LONG AS NOT TOP OF MEMORY
000013  275240  000700          SUBI    DAT,700         ;MAKE ROOM FOR TENDUMP AT TOP OF CORE
000014  264000  002141'          JSR MOVSTP              ;MOVE DDT SYMBOL TABLE POINTER TO LOWER CORE
                                                        ; (DDTSYM=36) WHERE EXEC DDT EXPECTS IT,
                                                        ; IF NOT ALREADY MOVED THERE, RETURN POINTER IN UUO
000015  574100  000014          HLRE TAC1,UUO           ;-LENGTH OF S. T.
                                                ;NOTE THAT THE SYMBOL TABLE POINTER IS MOVED FROM
                                                ; JOBSYM(JDAT) TO DDTSYM(JDAT) BY SYSMAK WHEN THE SYSTEM
                                                ; IS FIRST CREATED.
000016  322100  000034'          JUMPE TAC1,JRSTI1       ;0 IF NO S. T. TO MOVE UP
000017  213000  000002          MOVNS TAC1              ;+LENGTH
000020  550040  000014          HRRZ TAC,UUO            ;FIRST ADDRESS
000021  271042  000000          ADDI TAC,(TAC1)         ;LENGTH+FIRST ADDRESS
000022  504440  000002          HRL TAC,TAC1            ;XWD LENGTH,LENGTH+FIRST ADDRESS
000023  275241  000001          SUBI DAT,1(TAC)         ;NEW LAST+1-OLD LAST+1
000024  542240  000031'          HRRM DAT,ST01           ;DIST, TO MOVE
000025  200170  000014          MOVE TAC1,UUO           ;-N,FIRST ADD,
000026  271125  000001          ADDI TAC1,1(DAT)        ;FORM NEW S.T. POINTER
000027  202100  000000          MOVEM TAC1,DDTSYM       ;STORE IN LOWER CORE
000030  200121  777777          MOVE TAC1,-1(TAC)
000031  202121  000031' ST01:    MOVEM TAC1,.(TAC)
000032  274040  002317'          SUB TAC,[XWD 1,1]
000033  325040  000030'          JUMPGE TAC,.-3
```

```
                                EXTERN DEVLST,INTNUM,INTTAB
                        ;SETUP LOCATIONS 40 THRU 61

000034  200040  002320' JRSTI1: MOVE TAC,[XWD LOC40,40] ;SETUP LOWER CORE PI LOCATIONS
000035  251040  000061          PLT TAC,61

                        ;LINK DEVICE SERVICE ROUTINES TOGETHER - IF NOT LOADED BY BUILD

000036  332000  000000          SKIPE DEVLST            ;HAVE SERVICE ROUTINES ALREADY BEEN CHAINED TOGETHER?
000037  254000  000060'         JRST ALRCHN             ;YES, MUST HAVE BEEN LOADED WITH BUILD
000040  205040  000000          MOVSI TAC,INTNUM        ;NO, NEG. NO. OF SERVICE ROUTINES*2
000041  554241  000000  INTLOP: HLRZ DAT,INTTAB(TAC)    ;GET NEXT PI NUMBER
000042  405240  000007          ANDI DAT,7              ;MASK OUT NUMBER OF DDB'S
000043  242240  000001          LSH DAT,1               ;SHIFT LEFT ONE SO MATCH PI LOCATIONS
000044  322240  000056'         JUMPE DAT,NOPICH        ;DOES THIS DEVICE HAVE A PI CHANNEL (PTY)?
000045  200205  000040          MOVE ITEM,40(DAT)       ;YES, PICK UP JSR CH'N FROM LOWER CORE
000046  200240  000004  INTLP1: MOVE DAT,ITEM           ;SAVE IT IN DAT (EITHER A JRST DEV'NT OR
                                                        ; JEN @CH'N
000047  200205  000001          MOVE ITEM,1(DAT)        ;PICK UP INSTR, FOLLOWING INTERRUPT CONSO

000050  607200  000740          TLNN ITEM,000740        ;IS IT A JEN?
000051  254000  000046'         JRST INTLP1             ;NO, KEEP LOOKING
000052  200101  000041'         MOVE TAC1,INTTAB(TAC)   ;YES, LAST DEVICE SO FAR, GET DEV'NT
000053  505100  254000          HRLI TAC1,(JRST)        ;MAKE JRST INSTR.
000054  202105  000001          MOVEM TAC1,1(DAT)       ;CHANGE JEN @CH'N TO JRST DEV'NT
000055  202202  000001          MOVEM ITEM,1(TAC1)      ;MAKE DEV'NT+1 BE JEN @CH'N
000056  253040  000057' NOPICH: AOBJN TAC,.+1          ;PICKUP EVERY OTHER WORD?
000057  253040  000041'         AOBJN TAC,INTLOP        ;ANY MOVE INTERRUPT SERVICE ROUTINES?
000060                  ALRCHN:
                        IFN FTDISK,<

                        ;INITIALIZE DISK

                                EXTERN NULPDL,DSKINI
000060  201140  000000          MOVEI PDP,NULPDL
000061  260140  000000          PUSHJ PDP,DSKINI        ;INITIALIZE THE DISK
                                                        ; AFTER SETTING UP CHANNEL LOC
                                                        ; BUT BEFORE OVERWRITING ONCE WITH DDB'S
                        >
000062  200040  000066'         MOVE TAC,JRSTI
000063  202040  000001'         MOVEM TAC,LINKSR+1      ;DO ONCE ONLY
000064  336000  000036'         SKIPN DEVLST    ;HAVE DEVICE DATA BLOCKS BEEN LINKED?
000065  254000  000000          JRST LINKDB     ;NO, GO LINK DATA BLOCKS
                                                ;RETURN @LINKSR
                                                ;MULTIPLE DEVICE DATA BLOCKS ARE CREATED
                                                ;AS SPECIFIED BY SYSSIZ AND MAY ERASE PART
                                                ;OF ONCE ONLY CODE
000066  254020  000000' JRSTI:  JRST @LINKSR     ;YES,RETURN
```

```
                                EXTERN UUO0,ERROR,UUO2,LINKOR,DEVLST
                                EXTERN CH1,CH2,CH3,CH4,CH5,CH6,CH7

000067  000000  000000  LOC40:  0                       ;UUO PC
                        INTERNAL UUOTRP
000070  264000  000000  UUOTRP: JSR UUO0                ;TO UUO HANDLER
000071  264000  000000          JSR CH1
000072  265240  000000          JSP DAT,ERROR
000073  264000  000000          JSR CH2
000074  265240  000072'         JSP DAT,ERROR
000075  264000  000000          JSR CH3
000076  265240  000074'         JSP DAT,ERROR
000077  264000  000000          JSR CH4
000100  265240  000076'         JSP DAT,ERROR
000101  264000  000000          JSR CH5
000102  265240  000100'         JSP DAT,ERROR
000103  264000  000000          JSR CH6
000104  265240  000102'         JSP DAT,ERROR
000105  264000  000000          JSR CH7
000106  265240  000104'         JSP DAT,ERROR
000107  000000  000000          0                       ;LOC 60/61 TRAPPED TO FOR UNIMPLEMENTED OP CODES ON PDP-10
                                                        ;OP CODE 100 IS USED TO DISMISS USER MODE INTERRUPTS(SEE TRPSET
                                UUO)
000110  264000  000000          JSR     UUO2    ;GO SEE IF OP CODE 100 OR SOME OTHER ONE
```

```
                        IFN FT2REL,<
                        ;SET LOC TWOREG TO -1 IF MACHINE HAS TWO RELOC REG,0 IF ONY ONE RELOC REG
                                EXTERN TWOREG,SEGN,SEGPTR,AND2RG,JOBN

000111  000000  000000  TSTREG: 0
000112  200040  002321'          MOVE TAC,[JSR TSTAPR]   ;REPLACE UUO JSR WIT A JSR HERE IN ONCE
000113  202040  000041          MOVEM TAC,41
000114  700200  020001          CONO APR,ILM!1          ;TURN ON PI & ASSIGN ILM FLAG
                                                        ;FOR POSSIBLE ILLEGAL MEMORY REF BELOW
000115  202040  000042          MOVEM TAC,42            ;PUT A JSR IN CHAN 1 INT, LOC.
000116  700600  012300          CONO PI,12300           ;CLR PI SYSTEM, ACTIVATE PI SYSTEM, TURN ON CHAN 1
000117  700140  002322'          DATAO APR,[XWD 376776,000400]   ;SET LOW AND HIGH SEG TO 128K LONG
                                                        ; EACH STARTING AT ABS LOC 0
                                                        ; RELOC, FOR HIGH SEG IS 400000 TO DO THIS
000120  254040  000121'          JRST 1,.+1              ;ENTER USER MODE .+1
000121  200040  400100          MOVE TAC,400100         ;TRY TO REFERENCE REL LOC 100 IN HIGH SEG
                                                        ; AVOID SHADOW ACS WHICH LIKELY HAVE BAD
                                                        ;PARITY CAUSED BY POWER TURN ON AND TURN OFF
000122  201740  400000          MOVEI 17,400000         ;MOVE HARDWARE ACS INTO SHADOW ACS TO CLEAR PARITY
                                                        ;IF HARDWARE ACS TURNED OFF, ACS WILL BE PRESERVED
000123  251740  400017          BLT 17,400017           ;STOP WITH SHADOW AC 17
000124  040000  000000          CALL                    ;COME BACK TO EXEC MODE AT TSTAPR

000125  000000  000000  TSTAPR: 0                       ;PLACE TO STOE JSR PC
000126  402000  000000          SETZM TWOREG            ;ASSUME NO SECOND REG.
000127  700340  020000          CONSO APR,ILM           ;WAS MEMORY REF OUT OF BOUNDS?
000130  476000  000126'          SETOM TWOREG            ;NO, MUST HAVE SECOND REG. HARDWARE, SET -1
000131  700200  020000          CONO APR,ILM            ;CLEAR APR FLAGS AGAIN
000132  201040  000000          MOVEI TAC,AND2RG        ;=0 IF SOFTWARE CANNOT HANDLE 2 SEG(NULSEG LOADED)
000133  406040  000130'          ANDM TAC,TWOREG         ;=-1 IF IT CAN(SEGCON LOADED)
                                                        ;TWOREG SET TO AND OF SOFTWARE AND HARDWARE CAPABILITIE
                        S
000134  211040  000000          MOVNI TAC,SEGN          ;SET LH OF POINTER TO FIRST SEG IN IN JBTXXX TABLES
000135  505040  000000          HRLI TAC,JOBN           ;SET RH OF POINTER TO FIRST HIGH SEG
                                                        ;THIS MUST BE DONE HERE BECAUSE SEGPTR APPEARS IN FIR

                                                        ; WHICH BUILD FORBIDS FROM HAVING EXTERNALS
000136  206040  000000          MOVSM TAC,SEGPTR        ;POINTER USED FOR AOBJN LOOPS
000137  700600  010000          CONO PI,10000           ;CLEAR PI SYSTEM
000140  254020  000111'          JRST @TSTREG
                        >
```

```
                        ;ROUTINE TO FIND THE EXEC DDT SYMBOL TABLE POINTER AND MOVE IT TO THE PLACE
                        ;WHERE EXEC DDT EXPECTS IT(DDTSYM=36)
                        ;THE MONITOR CAN BE LOADED IN ANY OF 3 WAYS(IN ORDER OF PERFERENCE):
                        ;  1. UNDER TIME SHARING WITH REGULAR LOADER AND COMMON
                        ;  2. UNDER REGULAR 10/30 MONITOR WITH REGULAR 10/30 LOADER AND COMMON
                        ;  3. UNDER SPECIAL 10/30 MONITOR(SPMON) WITH BUILD
                        ;
                        ;THE 3 WAYS LEAVE DDTSYM(36),JOBSYM(116),T30SYM(131) IN DIFFERENT STATES:
                        ;       DDTSYM(36)      JOBSYM(116)     T30SYM(131)
                        ;  1.   JUNK            S,T,PTR         JUNK
                        ;  2.   JUNK            JUNK(NON-NEG)   S,T,PTR
                        ;  3.   S,T,PTR         S,T,PTR         JUNK
                        ;
                        ;CALL:  JSR MOVSTP
                        ;       RETURN WITH AC UUO SET TO SYMBOL TABLE POINTER
                        ;CALLED FROM MANDATORY ONCE ONLY CODE AND 141 START CODE(WHICHEVER OCCURS FIRST)

                                INTERN MOVSTP
                                EXTERN T30SYM,JOBSYM,DDTSYM,DDTX,SYSDDT

000141  000000  000000  MOVSTP: 0                       ;JSR HERE FROM MANDATORY ONCE ONLY CODE AND
                                                        ; FIRST JUMP TO EXEC DDT IF BEFORE MONITOR STARTED
000142  254000  000143' MOVJMP: JRST .+1               ;PATCHED TO SETUUO AFTER FIRST EXECUTION
000143  200600  000000          MOVE UUO,T30SYM         ;ASSUME LOADED BY REGULAR 10/30
000144  331000  000000          SKIPL JOBSYM            ;OR IS JOBSYM NEGATIVE(MEANING IT IS S,T,PTR)?
000145  254000  000151'         JRST REGT30             ;WAS LOADED BY REGULAR 10/30
000146  200600  000144'         MOVE UUO,JOBSYM         ;ASSUME LOADED BY TIME SHARING MONITOR AND LOADER
000147  332000  000064'         SKIPE DEVLST            ;OR HAS BUILD PATCHED THE DDB CHAIN?
000150  200600  000027'         MOVE UUO,DDTSYM         ;YES, DDTSYM HAS SYMBOL TABLE POINTER AS STORED
                                                        ; BY SPMON
000151  202600  000150' REGT30: MOVEM UUO,DDTSYM       ;STORE THE SYMBOL TABLE POINTER FOR EXEC DDT
000152  201040  000000          MOVEI TAC,DDTX          ;MAKE SYSTEM STARTUP LOC(141) GO DIRECTLY TO EXEC
                                                        ; DDT AND BY PASS THIS FOOLISHNESS
000153  542040  000000          HRRM TAC,SYSDDT
000154  201040  000156'         MOVEI TAC,SETUUO        ;MAKE SURE THIS IS DONE ONLY ONCE
000155  542040  000142'         HRRM TAC,MOVJMP
000156  200600  000151' SETUUO: MOVE UUO,DDTSYM         ;RETURN THE CORRECT(MAYBEUPDATED IF
                                                        ; EXEC DDT PATCHES MADE BEFORE MONITOR SAVED OR
                                                        ; STARTED) SYMBOL TABLE POINTER
000157  254020  000141'         JRST @MOVSTP            ;RETURN

                        ;HERE IF EXEC DDT IS STARTED UP BEFORE MONITOR(TO MAKE A PATCH FOR EXAMPLE)

                                INTERN PATSYM
                                EXTERN DDTX

000160  264000  000141' PATSYM: JSR MOVSTP              ;MOVE DDT SYMBOL TABLE POINTER SO EXEC DDT WILL
                                                        ; FIND IT AND PATCH 141 TO GO DIRECTLY TO EXEC DDT
000161  254030  000152'         JRST DDTX               ;AND GO DIRECTLY TO EXEC DDT
```

```
                                ;ONCE ONLY CODE - OPERATOR SETUP DIALOGUE


                                IFN     FTRC10, <
                                ENTRY   RCXWN2          ;THIS SYMBOL IS SOLELY TO PERMIT THE SYSTEM
000162                          RCXWN2:                 ; BUILDER TO RETRIEVE THE CORRECT BINARY FILE.
                                INTERNAL        ONCE
                                >
                                IFE     FTRC10, <
                                ENTRY ONCE
                                >

                                EXTERNAL CONMES,RADX10,CRLF,DECIN1,OCTPNT
                                EXTERNAL CONFIG,SYSDAT
                                INTERN FTTTYSER
                                IFE FTTTYSER,<
                                        DEFINE SETTYO <MOVE DAT,LINEP>
                                        DEFINE TYPE <IDPB CHREC,DAT>

                                        DEFINE NEXTC <ILDB CHREC,TAC>>
                                IFN FTTTYSER,<
                                        DEFINE SETTYO <PUSHJ PDP,OTSET>
                                        DEFINE TYPE <PUSHJ PDP,OUTCHS>
                                        DEFINE NEXTC <PUSHJ PDP,GETCHR>
                                EXTERN  TSETBF,SETBFI,SCNDDB,TTIBUF,TTOBUF,OUTCHS,GETCHR,TIBF,TOBF
                                >

                                        INTERN PATSIZ
                                        EXTERN PATCH

000162  000000  000000  ONCE:   0
000163  201040  000000  PATSIZ: MOVEI TAC,PATCH          ;SET SIZE OF SYSTEM TO BEGINING OF PATCH
                                MOVEM TAC,SYSSIZ        ;IN CASE ONCE ONLY DIALOG RESTARTED BEFORE OVERWRITTEN
000164  202240  000000
                                                        ;PATSIZ SHOULD BE UPDATED EVERY TIME A PATCH
                                                        ; IS MADE, SYSSIZ SHOULD ALSO BE UPDATED IF 146
                                                        ; START IS GOING TO BE USED(DDT DEBUGGING OR CTY DOWN)

000165  200140  002323'         MOVE PDP,[XWD -ONCEPD,ONCEPD]
000166  260140  001006'         SETTYO*PUSHJ PDP,OTSET*
000167  260140  000000          PUSHJ PDP,CRLF
000170  201040  000000          MOVEI TAC,CONFIG
000171  260140  000000          PUSHJ PDP,CONMES
000172  201040  002324'         MOVEI TAC,[ASCIZ / /]
000173  260140  000171'         PUSHJ PDP,CONMES
000174  201040  000000          MOVEI TAC,SYSDAT
000175  260140  000173'         PUSHJ PDP,CONMES
000176  201040  001077'         MOVEI TAC,TSEXEC
000177  260140  000175'         PUSHJ PDP,CONMES
000200  260140  000734'         PUSHJ PDP,OPOUT
```

```
                                ;ASK FOR TODAY'S DATE AND CONVERT
                                ;DATE STORED AS ((Y-64)*12,+M-1)*31,+D-1

                                EXTERNAL        THSDAT,JIFMIN,SKIPS

000201  260140  000203'          PUSHJ PDP,DATLOP
000202  254000  000201'          JRST .-1
000203  402000  000000  DATLOP: SETZM THSDAT
000204  201040  001104'          MOVEI TAC,TODATE
000205  260140  001002'          PUSHJ PDP,ICONM
000206  260140  000734'          PUSHJ PDP,OPOUT
000207  260140  000670'          PUSHJ PDP,GETLIN
000210  254000  000203'          JRST DATLOP
000211  260140  000000           PUSHJ PDP,SKIPS
000212  260140  000000           PUSHJ PDP,DECIN1          ;MONTH
000213  254000  000203'          JRST DATLOP
000214  254000  000203'          JRST DATLOP               ;ERROR
000215  332000  000002           SKIPE TAC1
000216  303100  000014           CAILE TAC1,+D12
000217  254000  000203'          JRST DATLOP
000220  275100  000001           SUBI TAC1,1
000221  221100  000037           IMULI TAC1,+D31
000222  272100  000203'          ADDM TAC1,THSDAT
000223  260140  000000           NEXTC+PUSHJ PDP,GETCHR+
000224  260140  000212'          PUSHJ PDP,DECIN1          ;DAY
000225  254000  000203'          JRST DATLOP
000226  254000  000203'          JRST DATLOP
000227  332000  000002           SKIPE TAC1
000230  303100  000037           CAILE TAC1,+D31
000231  254000  000203'          JRST DATLOP
000232  275100  000001           SUBI TAC1,1
000233  272100  000222'          ADDM TAC1,THSDAT
000234  260140  000223'          NEXTC+PUSHJ PDP,GETCHR+
000235  260140  000224'          PUSHJ PDP,DECIN1          ;YEAR
000236  254000  000203'          JRST DATLOP
000237  254000  000203'          JRST DATLOP
000240  301100  000105           CAIL TAC1,+D69
000241  303100  000143           CAILE TAC1,+D99
000242  254000  000203'          JRST DATLOP
000243  275100  000100           SUBI TAC1,+D64            ;YEAR ZERO
000244  221100  000564           IMULI TAC1,+D12*+D31
000245  272100  000233'          ADDM TAC1,THSDAT
```

```
                                ;GET TIME OF DAY

                                EXTERNAL        TIME

000246  262140  000001          POP PDP,TAC
000247  260140  000251'         PUSHJ PDP,TIMLOP
000250  254000  000247'         JRST .-1
000251  402000  000000  TIMLOP: SETZM TIME
000252  201040  001113'         MOVEI TAC,TIMEM
000253  260140  001002'         PUSHJ PDP,ICONM
000254  260140  000734'         PUSHJ PDP,OPOUT
000255  260140  000670'         PUSHJ PDP,GETLIN
000256  254000  000251'         JRST TIMLOP             ;JUST A CR
000257  260140  000211'         PUSHJ PDP,SKIPS
000260  260140  000235'         PUSHJ   PDP,DECIN1      ;READ THE TIME (1-4 CHARACTERS)
000261  254000  000251'         JRST    TIMLOP  ;NO ARGUMENT
000262  254000  000251'         JRST    TIMLOP  ;ILLEGAL CHARACTER
000263  200040  000002          MOVE    TAC,TAC1
000264  231040  000144          IDIVI   TAC,*D100       ;TAC=HRS,TAC1=MINS

000265  331000  000001          SKIPL   TAC     ;HRS NEGATIVE?
000266  303040  000027          CAILE   TAC,*D23        ;OR .G. 23?
000267  254000  000251'         JRST    TIMLOP  ;OUT OF BOUNDS
000270  303100  000073          CAILE   TAC1,*D59       ;MINUTES .G. 59
000271  254000  000251'         JRST    TIMLOP  ;OUT OF BOUNDS
000272  221040  000074          IMULI   TAC,*D60        ;CONVERT TO MINUTES
000273  270040  000002          ADD     TAC,TAC1        ;MINUTES SINCE MIDNIGHT
000274  221040  000000          IMULI   TAC,JIFMIN      ;JIFFIES SINCE MIDNIGHT
000275  202040  000251'         MOVEM   TAC,TIME        ;TIME=JIFFIES SINCE MIDNIGHT

000276  325200  000466'         JUMPGE  ITEM,QUICK      ;SKIP NEXT PART OF DIALOGUE UNLESS THE LAST ITEM
                                                        ; TYPED IN (NAMELY TIME OF DAY) WAS TERMINATED BY
                                                        ; AN "ALT MODE".  (ACCUMULATOR "ITEM" IS SET UP
                                                        ; IN THE "GETLIN" SUBROUTINE CALLED ABOVE.)
```

```
                        ;PRINT IO CONFIGURATION

                        EXTERNAL DEVLST,TCONLN,INLMES,PRNAME,INTTB1,INTTAB
                        EXTERNAL OCTPNT

                        IFN FT2REL,<
000277  264000  000111'          JSR TSTREG              ;TEST IF 2 RELOC REG SET TWOREG Ø OK-1
000300  201040  001200'          MOVEI TAC,CRLFMS        ;PRINT CRLF
000301  260140  001222'          PUSHJ PDP,ICONM
000302  201040  000001          MOVEI TAC,1             ;ASSUME 1
000303  332000  000133'          SKIPE TWOREG
000304  201040  000002          MOVEI TAC,2
000305  260140  000000          PUSHJ PDP,RADX10        ;NO, SECOND REG EXISTS
000306  260140  000000          PUSHJ PDP,INLMES
000307  202450  546236          ASCIZ / RELOC. REG.
000310  415344  051212
000311  435341  505000  /
000312  260140  000734'          PUSHJ PDP,OPOUT
                        >
000313  201040  001120'          MOVEI TAC,IOCONF
000314  260140  001222'          PUSHJ PDP,ICONM
000315  554640  000147'          HLRZ AC1,DEVLST
000316  326640  000340'          JUMPN AC1,ALRLNK        ;DEVICE DATA BLOCKS ALREADY LINKED?
000317  205340  000040'          MOVSI PROG,INTNUM       ;NO, NOT LOADED BY BUILD,-NO. OF DEVICES*2
000320  336647  000000  CONFLP: SKIPN AC1,INTTB1(PROG)  ;INT LOC HAVE DEV DATA BLOCK?
000321  254000  000335'          JRST NODDB              ;NO, GO GET NEXT DEVICE DATA BLOCK
000322  135600  002325'          LDB UUO,[POINT 8,INTTAB(PROG),8]        ;YES, NO. OF DEV DATA BLKS
000323  550040  000014          HRRZ TAC,UUO            ;PRINT
000324  260140  000335'          PUSHJ PDP,RADX10        ;IN DECIMAL
000325  260140  000326'          PUSHJ PDP,INLMES        ;PRINT 1 SPACE
000326  200000  000000          ASCIZ / /
000327  510115  000000          HLLZ TAC1,DEVNAM(AC1)   ;DEVICE NAME
000330  260140  000000          PUSHJ PDP,PRNAME        ;PRINT IT
000331  363620  000334'          SOJLE UUO,SINGLE                ;IS THIS JUST A SINGLE DEVICE?
000332  260140  000325'          PUSHJ PDP,INLMES        ;NO, APPEND 'S TO DEVICE NAME
000333  236460  000000          ASCIZ /'S/
000334  260140  000167' SINGLE: PUSHJ PDP,CRLF          ;PRINT CR LF
000335  253340  000336' NODDB:  AOBJN PROG,.+1          ;
000336  253340  000320'          AOBJN PROG,CONFLP       ;FINISHED ALL DEVICES?
000337  254000  000366'          JRST ONCE7A             ;YES
000340                  ALRLNK:
000340  201040  000001  ONCE5:  MOVEI TAC,1
000341  510215  000000          HLLZ ITEM,DEVNAM(AC1)
000342  554655  000003  ONCE4:  HLRZ AC1,DEVSER(AC1)
000343  322640  000347'          JUMPE AC1,ONCE6
000344  510115  000000          HLLZ TAC1,DEVNAM(AC1)
000345  316100  000004          CAMN TAC1,ITEM
000346  344040  000342'          AOJA TAC,ONCE4
000347  200340  000001  ONCE6:  MOVE PROG,TAC           ;SAVE NO.
000350  260140  000324'          PUSHJ PDP,RADX10
000351  201400  000040          MOVEI CHREC," "
000352  260140  000000          TYPE*PUSHJ PDP,OUTCHS*
000353  201000  000003          MOVEI UUO,3
```

```
431  000354  200040  002326'          MOVE TAC,[POINT 6,ITEM]
432  000355  134400  000001  ONCE7:   ILDB CHREC,TAC
433  000356  271400  000040          ADDI CHREC,40
434  000357  260140  000352'          TYPE*PUSHJ PDP,OUTCHS*
435  000360  367600  000355'          SOJG UU0,ONCE7
436  000361  201040  002327'          MOVEI TAC,[ASCIZ /'S/]
437  000362  303340  000001          CAILE PROG,1
438  000363  260140  000177'          PUSHJ PDP,CONMES
439  000364  260140  000334'          PUSHJ PDP,CRLF
440  000365  326640  000340'          JUMPN AC1,ONCE5
441  000366  260140  000734' ONCE7A: PUSHJ PDP,OPOUT
442  000367  201040  001200'          MOVEI TAC,CRLFMS
443  000370  260140  001002'          PUSHJ PDP,ICONM
444  000371  201040  000000          MOVEI TAC,TCONLN          ;PRINT NO, OF TTY SCANNER LINES
445  000372  260140  000000          PUSHJ PDP,OCTPNT
446  000373  260140  000332'          PUSHJ PDP,INLMES
447  000374  201211  741650          ASCIZ / (OCTAL) TTY SCANNER LINES
     000375  406305  120250
     000376  522624  051606

     000377  406351  642644
     000400  202311  147212
448  000401  514321  200000  /
449  000402  260140  000734'          PUSHJ PDP,OPOUT
```

```
450
451                             ;ASK FOR OPERATORS CONSOLE TO BE SPECIFIED
452
453                             EXTERNAL INLMES,DEVOPR,CTEXT,DEVPHY
454
455  000403  201040  001125' ONCEOP: MOVEI TAC,OPRM
456  000404  260140  001002'         PUSHJ PDP,ICONM
457  000405  260140  000734'         PUSHJ PDP,OPOUT
458  000406  260140  000670'         PUSHJ PDP,GETLIN
459  000407  254000  000417'         JRST ONCE8                  ;JUST CR
460  000410  260140  000000          PUSHJ PDP,CTEXT
461  000411  200040  000002          MOVE TAC,TAC1
462  000412  316040  002330'         CAMN TAC,[SIXBIT /CTY/]             ;CTY?
463  000413  254000  000416'         JRST .+3                            ;YES, JUST STORE IT.
464  000414  260140  000000          PUSHJ PDP,DEVPHY
465  000415  254000  000403'         JRST ONCEOP                 ;PHYSICAL NAME NOT FOUND
466  000416  202040  000000          MOVEM TAC,DEVOPR            ;SET FOR SYSTEM
467                                                              ;DEVICE "OPR" WILL BE SAME AS THIS
```

```
468
469                             ;ASK IF SYSMAK IS WANTED
470
471                                     EXTERN MAKEND,SYSSIZ
472
473  000417  201040  001133' ONCE8:   MOVEI TAC,SYSM
474  000420  260140  001015'          PUSHJ PDP,YESNO
475  000421  254000  000424'          JRST ONCE2              ;NO SYSMAK
476  000422  201040  000000           MOVEI TAC,MAKEND        ;YES
477  000423  202040  000164'          MOVEM TAC,SYSSIZ
478
479                             ;IS EXEC DDT WANTED?
480
481                             EXTERNAL        DDTEND
482
483  000424  201040  001145' ONCE2:   MOVEI TAC,SYSDM
484  000425  260140  001015'          PUSHJ PDP,YESNO
485  000426  254000  000431'          JRST ONCE3              ;NO
486  000427  201040  000000           MOVEI TAC,DDTEND        ;FIRST FREE LOC. ABOVE EXEC DDT
487  000430  202040  000423'          MOVEM TAC,SYSSIZ        ;SET NEW MONITOR SIZE
488                             ;FIND CORE TO BE USED FOR MULTIPLE DEVICE DATA BLOCKS
489                             ; INCLUDING DISK
490
491                             EXTERNAL CNTDB
492
493  000431  261140  000430' ONCE3:   PUSH PDP,SYSSIZ ;SAVE CURRENT VALUE
494  000432  265040  000000           JSP TAC,CNTDB   ;UPDATE SYSSIZ BY SPACE USED
495                                                   ; FOR DEVICE DATA BLOCKS FOR DSK,DTA,MTA,TTY,PTY)
496
```

```
                                ;PRINT OCTAL SIZE OF MONITOR

000433  201040  001150'          MOVEI TAC,EXECIS
000434  260140  001072'          PUSHJ PDP,ICONM
000435  200040  000431'          MOVE TAC,SYSSIZ
000436  260140  000372'          PUSHJ PDP,OCTPNT
000437  201040  001153'          MOVEI TAC,LENGTH
000440  260140  000363'          PUSHJ PDP,CONMES
000441  260140  000364'          PUSHJ PDP,CRLF
000442  260140  000734'          PUSHJ PDP,OPOUT
000443  262140  000435'          POP PDP,SYSSIZ  ;RESTORE SYSSIZ PRIOR TO ACTUAL CREATION
                                                ; OF DEVICE DATA BLOCKS
                                EXTERN DECIN,CORLIM,USRLIM,RADX10

000444  201040  001160' ONCE4A: MOVEI TAC,MXKMES         ;PRINT "MAX. K CORE FOR SINGLE USER IS "
000445  260140  001002'          PUSHJ PDP,ICONM
000446  201040  000000          MOVEI TAC,USRLIM
000447  260140  000350'          PUSHJ PDP,RADX10
000450  260140  000441'          PUSHJ PDP,CRLF
000451  260140  000734'          PUSHJ PDP,OPOUT

000452  201040  001167'          MOVEI TAC,DESMES         ;PRINT "TYPE DESIRED MAX.(DEC), CR IF OK AS IS
000453  260140  001002'          PUSHJ PDP,ICONM ;"
000454  260140  000734'          PUSHJ PDP,OPOUT
000455  260140  000670'          PUSHJ PDP,GETLIN
000456  254000  000465'          JRST ONCE5A              ;LEAVE LIMIT AS SET BY BUILDER QUESTION,
000457  260140  000257' IFN FTTTYSER,<PUSHJ PDP,SKIPS>
000460  260140  000000          PUSHJ PDP,DECIN
000461  254000  000444'          JRST ONCE4A
000462  254000  000444'          JRST ONCE4A
000463  332000  000002          SKIPE TAC1               ;UNLESS HE SAID 0 LIKE IN MONGEN,
000464  542100  000000          HRRM TAC1,CORLIM         ;PATCH CORLIM FOR USER CORE SIZE LIMIT
                                                         ; IF NOT PATCHED, LEAVE COMMON'S VALUE
000465                  ONCE5A:


                                INTERNAL        FTDISK
                                IFN FTDISK,<
000465  260140  000467'          PUSHJ PDP,REF            ;GO THROUGH THE REFRESH DIALOGUE.
                                >

000466                  QUICK:                  ;THIS IS THE SHORT-CUT EXIT TO BY-PASS MOST OF THE DIALOGUE.


                                INTERNAL        FTCHECK

                                IFN FTCHECK,<
                                        PUSHJ PDP,CK    ;CHECKSUM THE MONITOR AND SAVE THE ANSWER.
                                >

000466  254020  000162'          JRST @ONCE               ;***EXIT FROM THE "ONCE-ONLY CODE"***.....
```

```
                                IFN FTDISK,<
                                EXTERNAL OCTIN,MFDBLK,SATXWD,NUMBLK,SYSPP,REFLAG

                                ;REFRESH DIALOGUE

000467  201040  000555' REF:    MOVEI TAC,REF1M ;DO YOU WANT TO REFRESH?
000470  260140  001015'         PUSHJ PDP,YESNO
000471  263140  000000          POPJ PDP,               ;NO
000472  201040  000567'         MOVEI TAC,REF2M ;YES, ARE YOU SURE?
000473  260140  001015'         PUSHJ PDP,YESNO
000474  263140  000000          POPJ PDP,               ;NO
000475                          REF6:                   ;YES, DO YOU WANT TO CHANGE THE LOC
000475  201040  000574'         MOVEI TAC,REF3M         ;OF THE MFD?
000476  260140  001015'         PUSHJ PDP,YESNO
000477  254000  000514'         JRST REF3               ;NO
000500                          REF2:                   ;YES
000500  201040  000610'         MOVEI TAC,REF4M         ;TYPE LOGICAL DISK ADDRESS OF MFD RETR, INFO
000501  260140  001002'         PUSHJ PDP,ICONM
000502  260140  000734'         PUSHJ PDP,OPOUT

000503  260140  000670'         PUSHJ PDP,GETLIN
000504  254000  000500'         JRST REF2
000505  260140  000457'         IFN FTTTYSER,<PUSHJ PDP,SKIPS>
000506  260140  000000          PUSHJ PDP,OCTIN
000507  254000  000500'         JRST REF2
000510  254000  000500'         JRST REF2
                                IFE     FTRC10, <
                                PUSH PDP,TAC1           ;IS IT A MULTIPLE OF NUMBLK?
                                MOVEI TAC,(TAC1)
                                IDIVI TAC,NUMBLK
                                JUMPN TAC1,REF4         ;NO
                                POP PDP,TAC1
                                >
000511  552170  000000          HRRZM TAC1,MFDBLK
000512  350000  000002          AOS TAC1
000513  542170  000000          HRRM TAC1,SATXWD
```

```
585  000514                    REF3:
586                            INTERNAL FTSWAP
587                            IFN FTSWAP,<IFN FTRC10, <
588  000514  201040  000651'         MOVEI   TAC,REF9M       ;HOW MANY 1K BLOCKS ON THE DISK FOR SWAPPING ?
589  000515  260140  001002'         PUSHJ   PDP,ICONM
590  000516  260140  000734'         PUSHJ   PDP,OPOUT
591  000517  201040  000310          MOVEI   TAC,^D200       ;THE DEFAULT ANSWER IS 200 DECIMAL.
592  000520  202040  000000          MOVEM   TAC,K4SWAP
593  000521  260140  000670'         PUSHJ   PDP,GETLIN
594  000522  254000  000532'         JRST    REF3A           ;JUST A CARRIAGE RETURN INVOKES DEFAULT
595  000523  260140  000525'         IFN FTTTYSER,<PUSHJ PDP,SKIPS>
596  000524  260140  000460'         PUSHJ   PDP,DECIN
597  000525  254000  000514'         JRST    REF3
598  000526  254000  000514'         JRST    REF3
599  000527  303100  000000          CAILE   TAC1,MXK2SWP    ;ANSWER TOO BIG ?
600  000530  254000  000514'         JRST    REF3            ;YES, REPEAT QUESTION
601  000531  552100  000520'         HRRZM   TAC1,K4SWAP     ;NO, SAVE THIS CRUCIAL PARAMETER.
602  000532                    REF3A:
603                            >>
604
605
606
607  000532  260140  001315'         PUSHJ PDP,REFRES        ;REFRESH.
608  000533  254000  000545'         JRST REF5               ;ERROR
609
610
611  000534  201040  000625'         MOVEI TAC,REF6M         ;REFRESHING SUCCESSFULLY COMPLETED !
612  000535  260140  001002'         PUSHJ PDP,ICONM
613  000536  260140  000734'         PUSHJ PDP,OPOUT         ;TYPE "REFRESHED !"
614
615
616  000537  200040  000000          MOVE    TAC,SYSPP
617  000540  202040  000000          MOVEM   TAC,REFLAG      ;TO PERMIT AUTOMATIC LOG-IN UNDER
618                                                         ; PROJ-PROG NUMBER [1,1] EVEN
619  000541  201040  000636'         MOVEI   TAC,REF8M       ; THOUGH THE LOG-IN CUSP IS CLEARLY NOT
620  000542  260140  001002'         PUSHJ   PDP,ICONM       ; ON THE NEWLY REFRESHED DISK.
621  000543  260140  000734'         PUSHJ   PDP,OPOUT
622  000544  263140  000000          POPJ PDP,
```

```
                                IFE     FTRC10, <
                        REF4:   POP PDP,TAC1
                                MOVEI TAC,REF5M
                                PUSHJ PDP,ICONM
                                PUSHJ PDP,OPOUT
                                JRST REF2
                                >

000545  261140  000001  REF5:   PUSH PDP,TAC              ;ERROR,  ON COMING HERE, TAC HAS LOGICAL BLOCK
                                                         ; NUMBER WHICH COULD NOT BE WRITTEN
000546  201040  000630'         MOVEI TAC,REF7M
000547  260140  001002'         PUSHJ PDP,ICONM
000550  262140  000001          POP PDP,TAC
000551  260140  000436'         PUSHJ PDP,OCTPNT
000552  260140  000450'         PUSHJ PDP,CRLF
000553  260140  000734'         PUSHJ PDP,OPOUT
000554  254000  000475'         JRST REF6                ;GO BACK TO TRY AGAIN.


000555  422364  054636  REF1M:  ASCIZ /DO YOU WANT TO REFRESH THE DISK ? (Y OR CR)
000556  525012  740634
000557  521012  447500
000560  512130  651212
000561  516204  052220
000562  425010  444646
000563  455007  720120
000564  545011  751100
000565  416445  120032
000566  050000  000000  /
000567  406450  520262  REF2M:  ASCIZ /ARE YOU SURE(Y OR CR)?
000570  476524  051652
000571  512125  054520
000572  476444  041644
000573  245761  505000  /
000574  422364  054636  REF3M:  ASCIZ /DO YOU WANT TO CHANGE THE LOCATION OF THE MFD(Y OR CR)?
000575  525012  740634
000576  521012  447500
000577  416210  147216
000600  425012  444212
000601  202311  741602
000602  522231  747100
000603  476144  052220
000604  425011  543210
000605  242624  047644
000606  202072  224576
000607  064240  000000  /
```

```
  648  000610  522632  042500  REF4M:  ASCIZ /TYPE THE LOGICAL DISK ADDRESS OF THE MFD RETREIVAL INFORMATION
       000611  522210  523230
       000612  476171  141672
       000613  461010  444646
       000614  455010  142210
       000615  512132  351500
       000616  476144  052220
       000617  425011  543210
       000620  202450  552244
       000621  426232  647630
       000622  202231  643236
       000623  512330  152222
  649  000624  476341  505000  /
  650                           IFE     FTRC10, <
  651                           REF5M:  ASCIZ /MUST BE A MULTIPLE OF NUMBLK,
  652                           /
  653                           >
  654  000625  512130  651212  REF6M:  ASCIZ /REFRESHED !
       000626  516210  542120
  655  000627  204321  200000  /
  656  000630  426452  247644  REF7M:  ASCIZ /ERROR WHILE WRITING BLOCK /
       000631  202571  044632
       000632  425012  751222
       000633  522231  643500
       000634  412311  741626
       000635  200070  000000
  657  000636  522364  040652  REF8M:  ASCIZ   /TO AUTOMATICALLY LOG-IN UNDER [1,1], TYPE "LOGIN"
       000637  522371  540650
       000640  446070  146230
       000641  545011  447616
       000642  266231  620252
       000643  472110  551100
       000644  555425  430672
       000645  261012  454640
       000646  425004  246236
       000647  436231  621032
  658  000650  050000  000000  /
  659                           IFN     FTRC10, <
  660  000651  442372  720232  REF9M:  ASCIZ   /HOW MANY (DECIMAL) 1K DISK BLOCKS ARE TO BE ALLOCATED FOR SWAPPING ?
       000652  406353  120120
       000653  422130  344632
       000654  406325  120142
       000655  455010  444646
       000656  455010  246236
       000657  416272  320202
       000660  512124  052236
       000661  202050  520272
       000662  462311  741602
       000663  522130  420214
       000664  476444  051656
       000665  406412  044634
  661  000666  435007  706424  /
       000667  000000  000000
```

```
662                             >
663                             >               ;THIS CLOSES AN FTDISK CONDITIONAL SEVERAL PAGES BACK.
```

```
                                ;ROUTINE TO READ A LINE FROM OPERATORS CONSOLE
                                ;CALL:  PUSHJ PDP,GETLIN
                                ;       JUST A CR TYPED IN
                                ;       A LINE TYPED IN,TAC SET AS BYTE POINTER

                        000010  CHREC=TEM               ;CHAR TO TYPE OUT (MUST BE SAME AS SCNSER ROUTINE)
                        000002  LINE=TAC1               ;TTY LINE NO. (ALSO SAME AS SCNSER)

                                IFN FTTTYSER,<EXTERN TSETBF,SETBFI,SCNDDR,DCPUTR,PUTCHI>

                                EXTERNAL        CPOPJ1

000670                          GETLIN: IFE FTTTYSER,<MOVE TAC,LINEP>
000670  260140  000000                  IFN FTTTYSER,< PUSHJ PDP,SETBFI>
000671  403200  000010                  SETZB ITEM,CHREC
000672  260140  000771'         GET1:   PUSHJ PDP,TYI
000673  712040  000010                  DATAI TTY,CHREC
000674  405400  000177                  ANDI CHREC,177
000675  306400  000177                  CAIN CHREC,177          ;RUBOUT?
000676  254000  000717'                 JRST DELETE
000677  306400  000033                  CAIN    CHREC,33                ;ALTMODE ?
000700  254000  000731'                 JRST    GETLN1          ;YES
000701  302400  000175                  CAIE    CHREC,175       ;VARIATION OF ALT-MODE ?
000702  306400  000176                  CAIN    CHREC,176       ;OR YET THIS THIRD VARIATION ?
000703  254000  000731'                 JRST    GETLN1          ;YES
000704  260140  000751'         GET2:   PUSHJ PDP,TYO           ;ECHO CHARACTER
000705  301400  000140                  CAIL    CHREC,140       ;IS THIS LOWER CASE?
000706  620400  000040                  TRZ     CHREC,40                ;YES,MAKE IT UPPER CASE
                                IFE FTTTYSER,<IDPB CHREC,TAC>
000707  260140  000000          IFN FTTTYSER,<PUSHJ PDP,PUTCHI  ;STORE CHARACTER
000710  255000  000000                  JFCL                    ;IGNORE ERROR RETURN
                                >
000711  302400  000015                  CAIE CHREC,15
000712  344200  000672'                 AOJA ITEM,GET1
000713  201400  000012                  MOVEI CHREC,12          ;OUTPUT LF
000714  260140  000751'                 PUSHJ PDP,TYO           ;TYPE OUT
                                IFE FTTTYSER,<  MOVE TAC,LINEP          ;RESET BYTE POINTER>
000715  326200  000000                  JUMPN ITEM,CPOPJ1       ;NULL LINE?
000716  263140  000000                  POPJ PDP,               ;YES, DON'T SKIP RETURN

000717  260140  000000          DELETE: IFN FTTTYSER,< PUSHJ PDP,DCPUTR
000720  254000  000724'                 JRST DELET1>
                                IFE FTTTYSER,<
                                        JUMPE ITEM,DELET1               ;AT BEGINNING OF LINE?
                                        ADD TAC,[XWD 070000,0]          ;NO, BACK UP BYTE POINTER
                                        TLNE TAC,400000                 ;FINISHED THIS WORD YET?
                                        ADD TAC,[XWD 347777,-1]         ;YES, BACK UP ADDRESS>
000721  201400  000134                  MOVEI CHREC,134                 ;OUTPUT BACK SLASH
000722  260140  000751'                 PUSHJ PDP,TYO
000723  364200  000672'                 SOJA ITEM,GET1

000724  201400  000015          DELET1: MOVEI CHREC,15
000725  260140  000751'                 PUSHJ PDP,TYO
```

```
  717  000726  201400  000012           MOVEI CHREC,12
  718  000727  260140  000751'          PUSHJ POP,TYO
  719  000730  254000  000670'          JRST GETLIN
  720
  721  000731  476000  000004  GETLN1: SETOM   ITEM            ;MECHANISM USED TO BY-PASS PART OF DIALOGUE
  722  000732  201400  000015           MOVEI   CHREC,15              ; WHEN ALTMODE IS TYPED.
  723  000733  254000  000704'          JRST    GET2
```

```
724                             ;ROUTINE TO TYPE A LINE ON OPERATOR CONSOLE
725                             ;ECHO CHECK STOPS LINE AND RETURNS
726                             ;CALL:  DAT SET TO END OF MESSAGE
727
728                             IFN FTTTYSER,<EXTERNAL CPOPJ>
729
730  000734                     OPOUT:  IFE FTTTYSER,<MOVE TAC,LINEP>
731  000734  201246  000000             IFN FTTTYSER,<MOVEI DAT,TTOBUF(DEVDAT)>
732  000735  712200  001000             CONO TTY,1000                   ;CLEAR INPUT
733                             OPOUT1: IFE FTTTYSER,<
734                                     CAMN TAC,DAT
735                                     POPJ PDP,>
736  000736  712340  000040             CONSZ TTY,40
737  000737  254000  000744'             JRST OPOUT2                     ;ECHO CHECK
738  000740  260140  000234'             NEXTC←PUSHJ PDP,GETCHR←
739  000741  322400  000000  IFN FTTTYSER,<  JUMPE CHREC,CPOPJ>
740  000742  260140  000751'             PUSHJ PDP,TYO
741  000743  254000  000736'             JRST OPOUT1
742
743  000744  201400  000015  OPOUT2: MOVEI CHREC,15
744  000745  260140  000751'             PUSHJ PDP,TYO
745  000746  201400  000012             MOVEI CHREC,12
746  000747  260140  000751'             PUSHJ PDP,TYO
747  000750  712200  001000             CONO TTY,1000                   ;CLEAR INPUT
```

```
                                ;WAIT TILL OUTPUT BUSY OFF BEFORE TYPING OUT CHAR.

                                EXTERNAL TCONLN,TYPL

000751  260140  000775' TYO:    PUSHJ PDP,APRCHK        ;CHECK CLOCK
000752  712300  000020          CONSZ TTY,20
000753  254000  000751'         JRST TYO
000754  261140  000001          PUSH PDP,TAC
000755  261140  000002          PUSH PDP,LINE
000756  200040  000010          MOVE TAC,CHREC
000757  221040  200401          IMULI TAC,200401
000760  404040  002331'         AND TAC,[OCT 11111111]
000761  220040  002331'         IMUL TAC,[OCT 11111111]
000762  603040  000010          TLNE TAC,10
000763  660400  000200          TRO CHREC,200
000764  712140  000010          DATAO TTY,CHREC
000765  620400  000200          TRZ CHREC,200
000766  262140  000002          POP PDP,LINE
000767  262140  000001          POP PDP,TAC
000770  263140  000000          POPJ PDP,

                                ;WAIT TIL INPUT DONE ON BEFORE RETURNING WITH NEXT CHAR.

000771  260140  000775' TYI:    PUSHJ PDP,APRCHK
000772  712340  000040          CONSO TTY,40
000773  254000  000771'         JRST TYI
000774  263140  000000          POPJ PDP,

                                ;CHECK APR FOR CLOCK FLAG SO TIME USER TYPES IN WILL BE ACCURATE

000775  700340  001000  APRCHK: CONSO APR,1000          ;IS CLOCK FLAG ON?
000776  263140  000000          POPJ PDP,               ;NO
000777  700200  001000          CONO APR,1000           ;YES, CLEAR CLOCK FLAG
001000  350000  000275'         AOS TIME                ;INCREMENT TIME
001001  263140  000000          POPJ PDP,
```

```
784  001002  261140  000001  ICONM:  PUSH    P,TAC
785                               SETTYO  ;INITIALIZE LINE BUFFER
786  001003  260140  001006' PUSHJ P,OTSET
787  001004  262140  000001          POP     P,TAC
788  001005  254000  000440'         JRST    CONMES  ;OUTPUT MESSAGE
789
790                           IFN FTTTYSER,<
791  001006  201300  000000  OTSET: MOVEI   TEM,SCNDDB
792  001007  205400  000120          MOVSI   TEM,120 ;TTYCHR
793  001010  541406  000000          HRRI    TEM,TIBF(DEVDAT)
794  001011  202406  000000          MOVEM   TEM,TTIBUF(DEVDAT)      ;INIT INPUT POINTER
795  001012  541406  000000          HRRI    TEM,TOBF(DEVDAT)        ;AND OUTPUT POINTER
796  001013  202406  000734'        MOVEM   TEM,TTOBUF(DEVDAT)
797  001014  254000  000000          JRST    TSETBF          ;NOW CLEAR THE BUFFERS
798                           >
799
800  001015  260140  001002' YESNO:  PUSHJ   P,ICONM        ;FIRST OUTPUT THE QUESTION
801  001016  260140  000734'         PUSHJ   P,OPOUT
802  001017  260140  000670'         PUSHJ   P,GETLIN        ;NOW GET RESPONSE
803  001020  263140  000000          POPJ    P,0     ;JUST C-R
804  001021  260140  000740'         NEXTC:PUSHJ P,GETCHR:
805  001022  620400  000040          TRZ     CHREC,40        ;FIRST CHAR OF RESPONSE (U.C.)
806  001023  306400  000131          CAIN    CHREC,"Y"       ;WAS IT A Y?
807  001024  350003  000000          AOS     0(P)    ;YES, SKIP
808  001025  263140  000000          POPJ    P,      ;NO, MAYBE IT WAS "N", SO DONT SKIP
```

```
809
810                                     ;CONSTANTS AND PUSH DOWN LIST
811
812                             000020  ONCEPN=20
813  001026                             ONCEPD: BLOCK ONCEPN               ;PUSH DOWN LIST
814
815  001046  440700  001047'            LINEP:  POINT 7,LINBUF             ;INPUT AND OUTPUT LINE BUFFER
816  001047                             LINBUF: BLOCK 30
817
818                                     ;MESSAGES
819
820  001077  202331  747222            TSEXEC: ASCIZ / MONITOR JUST LOADED
     001100  522372  220224
     001101  526472  420230
     001102  476030  442610
821  001103  064240  000070  /
822  001104                             TODATE: ASCIZ /
823  001104  064252  454640            TYPE TODAY'S DATE AS MM-DD-YY
     001105  425012  447610

     001106  406624  751500
     001107  422032  442500
     001110  406464  046632
     001111  266110  426662
824  001112  544321  200000  /
825
826  001113                             TIMEM:  ASCIZ /
827  001113  064252  454640            TYPE TIME AS HHMM
     001114  425012  444632
     001115  425010  151500
     001116  442211  546432
828  001117  050000  000000  /
829  001120                             IOCONF: ASCIZ /
830  001120  064251  147500            IO CONFIGURATION
     001121  416371  643222
     001122  436532  240650
831  001123  446371  606424  /
     001124  000000  000000
832
833  001125                             OPRM:   ASCIZ /
834  001125  064252  454640            TYPE NAME OF OPR CONSOLE
     001126  425011  640632
     001127  425011  743120
     001130  476412  220236
     001131  476352  347630
835  001132  424321  200000  /
```

```
836  001133                    SYSM:   ASCIZ /
837  001133  064250  447500    DO YOU WANT SYSMAK (TYPE Y IF YES, CR IF NO)?
     001134  546372  520256
     001135  406352  420246
     001136  546471  540626
     001137  201212  454640
     001140  425013  120222
     001141  431013  142646
     001142  261010  351100
     001143  446144  047236
838  001144  245761  505000    /
839  001145                    SYSDM:  ASCIZ /
840  001145  064250  554212    EXEC DDT?
     001146  415010  442250
841  001147  374321  200000    /
842  001150                    EXECIS: ASCIZ /
843  001150  064250  554212    EXEC IS /
     001151  415011  151520
     001152  000000  000000
844  001153  202370  352202    LENGTH: ASCIZ / OCTAL LOCATIONS LONG.
     001154  461011  447636
     001155  406511  147634
     001156  515011  447634
845  001157  435341  505000    /
846  001160  466033  020226    MXKMES: ASCIZ /MAX K CORE FOR SINGLE USER IS /
     001161  202071  751212
     001162  202151  751100
     001163  516231  643630
     001164  425012  551612
     001165  511011  151500
     001166  000000  000000
847  001167  522632  042500    DESMES: ASCIZ /TYPE DESIRED MAX,(DEC), CR IF OK AS IS
     001170  422132  344644
     001171  426104  046602
     001172  541345  042212
     001173  415225  420206
     001174  511011  143100
     001175  476264  040646
848  001176  202232  306424    /
     001177  000000  000000
849  001200                    CRLFMS: ASCIZ /
850  001200  064240  000000    /
851
```

```
852                             ;INITIALIZE SAT SEARCH ENTRIES
853
854                             ;THIS "DFWUNS" ROUTINE IS CALLED BY MANDATORY ONCE ONLY CODE(LINKSR)
855                             ;  AFTER ALL MULTIPLE DEVICE DATA BLOCKS HAVE BEEN GENERATED IN CORE
856                             ;  ABOVE C(SYSSIZ)( AND SYSSIZ HAS BEEN INCREMENTED)
857                             ;     IT IS CALLED WHETHER OR NOT THE DISK WAS REFRESHED.  IF REFRESHING WAS
858                             ;  PERFORMED, DFWUNS IS CALLED AFTER REFRES SINCE SYSINI CALLS LINKSR AFTER ONCE.
859
860                             INTERNAL FTDISK
861                             IFN FTDISK,<
862
863                             INTERNAL DFWUNS
864                             EXTERNAL SAT05,JSAT06,SATENT,SATPTR,SATBK2,SATXWD,SATPTR
865                             EXTERNAL SAT,SENTSZ,DSKDDB,DSKBUF,WSYNC,MQIN,MQOUT,DFGETF,SETFRE
866                             EXTERNAL WLBIT,NUMBIT,DSKCNT
867
868  001201                     DFWUNS:
869                             IFN     FTRC10, <
870  001201  260140  001723'          PUSHJ   P,DP,WRITLK       ;FORCE THE WRITE-LOCK SWITCH SETTINGS  TO BE ZERO.

871  001202  260140  001625'          PUSHJ   P,DP,RDSTAT       ;READ VITAL DISK STATISTICS BLOCK, STORE PARAMETERS
872  001203  260140  001660'          PUSHJ   P,DP,CAPCTY       ;DETERMINE CURRENT SYSTEM CAPACITY (HOW MANY DISKS).
873  001204  316040  000000           CAMN    T,AC,LBHIGH       ;HAS CAPACITY CHANGED SINCE LAST REFRESH ?
874  001205  254020  001241'          JRST    DFWNZ1            ;NO, EVERYTHING APPPEARS TO BE OKAY.
875                                     SETTYO  <PUSHJ P,DP,OTSET>                      ;YES, PRINT WARNING MES
876  001206  260140  001006' SAGE....
877  001207  260140  000373'          PUSHJ   P,DP,INLMES
878  001210  422232  345500           ASCIZ   /DISK CAPACITY HAS CHANGED SINCE LAST REFRESHING.
       001211  416032  040606
       001212  446513  120220
       001213  406464  041620
       001214  406350  742610
       001215  202471  147206
       001216  425011  440646
       001217  521012  242614
       001220  512132  344222
879  001221  472165  606424  EITHER RESTORE PREVIOUS DISK CONFIGURATION OR REFRESH.
       001222  426232  444212
       001223  511012  242646
       001224  522372  242500
       001225  502450  553222
       001226  476532  320210
       001227  446471  320206
       001230  476350  644616
       001231  526450  152222
       001232  476344  047644
       001233  202450  543244
       001234  426471  027032
880
881  001235  050321  200000  /
882  001236  260140  000734'          PUSHJ   P,DP,OPOUT        ;START TTY
883  001237  260140  000467'          PUSHJ   P,DP,REF          ;GO THROUGH THE REFRESH DIALOGUE
884  001240  254000  001201'          JRST    DFWUNS            ;NOW TRY AGAIN.
885  001241                     DFWNZ1:
```

```
886                                     >
887  001241  201040  000000                 MOVEI TAC,SATENT
888  001242  202040  000000                 MOVEM TAC,SATPTR
889  001243  201340  000000                 MOVEI PROG,0
890  001244  201370  000000                 MOVEI DEVDAT,DSKDDB
891  001245  200100  000000                 MOVE TAC1,SATBK2
892  001246  202106  000000                 MOVEM TAC1,DSKBUF(DEVDAT)
893  001247  261140  000000                 PUSH PDP,WSYNC
894  001250  200040  000000                 MOVE TAC,JSAT06
895  001251  202040  001247'                MOVEM TAC,WSYNC
896  001252  334040  000513'                SKIPA TAC,SATXWD
897
898                                     ;READ EACH SAT BLOCK FROM THE DISK AND SET SATENT TABLE TO MATCH INFO ON DISK.
899
900  001253  272100  001242' SETS1:  ADDM TAC1,SATPTR
901  001254  261140  000001  SETS2:  PUSH PDP,TAC
902  001255  403006  000002          SETZR IOS,DEVIOS(DEVDAT)
903  001256  260140  000000          PUSHJ PDP,MQIN              ;READ A SAT BLOCK
904  001257  254200  001313'        JRST 4,SETS6                ;ERROR HALT
905  001260  200240  001253'        MOVE DAT,SATPTR             ;RESET THE ENTRY
906  001261  200040  001245'        MOVE TAC,SATBK2
907  001262  202045  000002          MOVEM TAC,2(DAT)
908  001263  205100  400000          MOVSI TAC1,400000
909  001264  202145  000001          MOVEM TAC1,1(DAT)
910  001265  201240  000000          MOVEI DAT,0
```

```
911  001266  200641  000000  SETS3:  MOVE AC1,(TAC)          ;COUNT ONE-BITS
912  001267  346640  001272'          AOJN AC1,SETS3A         ;IS THE WORD = -1?
913  001270  271240  000044          ADDI DAT,44             ;YES
914  001271  254000  001277'          JRST SETS5
915
916  001272  362640  001277' SETS3A: SOJE AC1,SETS5          ;NO, IS IT = 0?
917  001273  612101  000000  SETS4:  TDNE TAC1,(TAC)         ;NO, COUNT THE BITS
918  001274  271240  000001          ADDI DAT,1
919  001275  241100  777777          ROT TAC1,-1
920  001276  325120  001273'          JUMPGE TAC1,SETS4
921
922  001277  253040  001266' SETS5:  AOBJN TAC,SETS3         ;LOOP THRU ENTIRE BLOCK
923  001300  542260  001260'          HRRM DAT,@SATPTR
924
925                           IFE     FTRC10, <
926                           ;BITS HAVE BEEN COUNTED, CHECK FOR WRITE-LOCK
927                                   CAIL DAT,NUMBIT         ;BLOCK FULL?
928                                   JRST SETS8
929                                   PUSHJ PDP,DFGETF        ;NO, GET A FREE BLOCK

930                                   PUSHJ PDP,SETS7         ;WRITE
931                                   PUSHJ PDP,SETFRE        ;FREE THE BLOCK
932                           >
933
934
935  001301  201100  000000  SETS5B: MOVEI TAC1,SENTSZ       ;PREPARE TO BUMP SATPTR
936  001302  262140  000001          POP PDP,TAC
937  001303  253040  001253'          AOBJN TAC,SETS1         ;DO ALL SAT BLOCKS
938
939
940                           IFE     FTRC10, <
941                                   JRST SAT05              ;RETURN
942                           >
943
944
945                           IFN     FTRC10, <
946  001304  550040  001300'          HRRZ    TAC,SATPTR
947  001305  301040  000000  SETS5X: CAIL    TAC,SATTOP
948  001306  254000  000000          JRST    SAT05           ;EXIT FROM DFWUNS BACK TO DSKINI IN DSKSER.....
949  001307  271040  001301'          ADDI    TAC,SENTSZ
950  001310  201100  411000          MOVEI   TAC1,411000     ;ON THE NEW PDP-10 DISK SYSTEM SET THE SAT ENTRY
951  001311  542101  000000          HRRM    TAC1,(TAC)      ; TABLE TO SHOW ALL NON-EXISTENT DISKS AS
952  001312  254000  001305'          JRST    SETS5X          ; BOTH WRITE-LOCKED AND FULL.
953                           >
```

```
954
955  001313  262140  000001  SETS6:  POP PDP,TAC             ;ERROR, TRY AGAIN
956  001314  254000  001254'          JRST SETS2
957
958
959                          IFE     FTRC10, <
960
961                          ;WRITE-LOCK DETERMINATION ON THE OLD PDP-6 (DATA PRODUCTS) DISK --
962
963                          SETS7:  PUSH PDP,TAC            ;SAVE BLOCK ADDRESS
964                          SETS7A: PUSHJ PDP,MQOUT
965                                  JRST SETS7C             ;ERROR
966                          SETS7B: POP PDP,TAC
967                                  POPJ PDP,
968                          SETS7C: MOVE TAC,(PDP)
969                                  MOVE TAC1,DSKCNT(DEVDAT)
970                                  TRNN TAC1,IOIMPM        ;WRITE LOCK?
971                                  JRST 4,SETS7A           ;NO
972                                  MOVEI TAC,WLBIT         ;YES, SET BIT
973                                  ORM TAC,@SATPTR
974                                  JRST SETS7B
975
976                          SETS8:  MOVEI TAC,SETS9         ;BLOCK FULL, READ FIRST ONE
977                                  HRRM TAC,DSKBUF(DEVDAT)
978                                  HLRZ TAC,@SATPTR
979                                  PUSHJ PDP,MQIN
980                                  JRST 4,.-2              ;ERROR
981                                  HLRZ TAC,@SATPTR        ;WRITE IT BACK
982                                  PUSHJ PDP,SETS7
983                                  MOVEI TAC,SAT           ;RESET
984                                  HRRM TAC,DSKBUF(DEVDAT)
985                                  JRST SETS5B
986
987
988                          SETS9:  BLOCK 204
989
990                          >               ;CLOSE THE IFE FTRC10
991                          >               ;CLOSES AN IFN FTDISK A COUPLE PAGES BACK.
```

```
                        IFN FTDISK,<

                        ;DISK REFRESHING ROUTINES ---

                        IFE     FTRC10, <
                        ;DISK REFRESHER FOR THE OLD PDP-6 (DATA PRODUCTS) DISK --


                        EXTERNAL MFDBLK,NUMSAT,CHKCNT,CHKSUM,NUMBLK,DSKINI,SETSAT,DSKCHN,DSKCHL

                        REFRESH:        MOVEI PROG,0            ;SETUP IO
                                MOVEI DEVDAT,DSKDDB
                                PUSH PDP,SETSAT
                                MOVE TAC,RPOPJ
                                MOVEM TAC,SETSAT
                                PUSHJ PDP,DSKINI
                                POP PDP,SETSAT
                                MOVEI TAC,DSKCHN
                                LSH TAC,1

                                ADDI TAC,40
                                MOVE TAC1,[JSR DSKCHL]
                                MOVEM TAC1,(TAC)
                                PUSH PDP,WSYNC
                                MOVE TAC,JSAT06
                                MOVEM TAC,WSYNC

                        ;SETUP NULL SAT BLOCK

                                SETOM DAT
                                MOVE TAC1,[XWD NULBLK,NULBLK+1]
                                SETZM NULBLK
                                BLT TAC1,NULBLK+177
                                MOVEI TAC,NUMBLK                ;END OF BLOCK BYTE POINTER
                                IDIVI TAC,^D36
                                SUBI TAC1,^D36
                                MOVMS TAC1
                                ROT TAC1,-6
                                ADDI TAC,NULBLK
                                HLL TAC,TAC1
                                TLOA TAC,100
                                IDPB DAT,TAC
                                TLNE TAC,770000
                                JRST .-2
                                AOS TAC
                                MOVEM DAT,(TAC)
                                HRLS TAC
                                AOS TAC
                                BLT TAC,NULBLK+177

                        ;SET UP SAT BLOCK WITH MFD AND SAT BLOCKS ALLOCATED

                                MOVE TAC,[XWD NULBLK,SATBL1]
                                BLT TAC,SATBL1+177
```

```
1045                                    MOVE TAC,[POINT 1,SATBL1]
1046                                    MOVEI TAC1,NUMSAT
1047                                    HRLM TAC1,SATRET+3
1048                                    ADDI TAC1,3
1049                                    IDPB DAT,TAC
1050                                    SOJG TAC1,.-1
1051
1052                            ;COMPUTE CHECKSUMS IF REQUIRED
1053
1054                                    MOVEI DAT,CHKCNT
1055                                    JUMPE DAT,REFR1
1056                                    MOVE TAC,[XWD -200,MFD1]
1057                                    PUSHJ PDP,CHKSUM
1058                                    HRLM TAC1,MFDRET+4
1059                                    MOVE TAC,[XWD -200,NULBLK]
1060                                    PUSHJ PDP,CHKSUM
1061                                    HRLM TAC1,SATRET+4
1062                                    MOVE TAC,[XWD -200,SATBL1]
1063                                    PUSHJ PDP,CHKSUM
1064                                    MOVE DAT,TAC1
1065
1066                            ;RELOCATE POINTERS RELATIVE TO MFDBLK
1067
1068                            REFR1:  MOVE TAC,MFDBLK
1069                                    HRRM TAC,MFDRET+177
1070                                    HRRM TAC,MFD1+1
1071                                    ADDI TAC,NUMSAT
1072                                    AOS TAC
1073                                    HRRM TAC,SATRET+177
1074                                    HRRM TAC,MFD1+3
1075                                    AOS TAC
1076                                    HRRM TAC,MFDRET+4
1077                                    MOVNI TAC1,NUMSAT
1078                                    MOVSS TAC1
1079                                    HRRI TAC1,1
1080                                    MOVE TAC,MFDBLK
1081                                    HLL TAC,SATRET+4
1082                                    HRREM TAC1,SATRET+3(TAC1)
1083                                    ADDM TAC,SATRET+3(TAC1)
1084                                    AOBJN TAC1,.-2
1085
1086                            ;FIX CHECKSUM FOR SATBL1
1087
1088                                    MOVE TAC,MFDBLK
1089                                    IDIVI TAC,NUMBLK
1090                                    HRLM DAT,SATRET+3(TAC)
1091                                    MOVE DAT,TAC
1092
1093                            ;WRITE BLOCKS ON DISK
1094
1095                                    MOVE TAC,MFDBLK         ;MFD RETREIVAL INFORMATION
1096                                    MOVE TAC1,[XWD -200,MFDRET]
1097                                    MOVEM TAC1,DSKBUF(DEVDAT)
```

```
1098                                    SETZB IOS,DEVIOS(DEVDAT)
1099                                    PUSHJ PDP,MQOUT
1100                                    JRST REFER              ;ERROR
1101                                    MOVEI TAC,1
1102                            REFR2:  MOVE TAC1,[XWD -200,MULBLK]     ;SAT BLOCKS
1103                                    CAMN TAC,DAT
1104                                    MOVE TAC1,[XWD -200,SATBL1]
1105                                    PUSH PDP,TAC
1106                                    ADD TAC,MFDBLK
1107                                    SETZB IOS,DEVIOS(DEVDAT)
1108                                    MOVEM TAC1,DSKBUF(DEVDAT)
1109                                    PUSHJ PDP,MQOUT
1110                                    JRST REFER1             ;ERROR
1111                                    POP PDP,TAC
1112                                    CAIGE TAC,NUMSAT
1113                                    AOJA TAC,REFR2
1114                                    AOS TAC                 ;SAT RETREIVAL INFORMATION
1115                                    ADD TAC,MFDBLK
1116                                    SETZB IOS,DEVIOS(DEVDAT)

1117                                    MOVE TAC1,[XWD -200,SATRET]
1118                                    MOVEM TAC1,DSKBUF(DEVDAT)
1119                                    PUSHJ PDP,MQOUT
1120                                    JRST REFER              ;ERROR
1121                                    MOVE TAC,MFDBLK         ;MFD
1122                                    ADDI TAC,NUMSAT
1123                                    ADDI TAC,2
1124                                    MOVE TAC1,[XWD -200,MFD1]
1125                                    MOVEM TAC1,DSKBUF(DEVDAT)
1126                                    SETZB IOS,DEVIOS(DEVDAT)
1127                                    PUSHJ PDP,MQOUT
1128                                    JRST REFER              ;ERROR
1129                                    POP PDP,WSYNC
1130                                    JRST CPOPJ1
1131
1132                            REFER1: POP PDP,TAC1
1133                            REFER:  POP PDP,WSYNC
1134                            RPOPJ:  POPJ PDP,
1135
1136                            DEFINE ZEROES (A),<XLIST
1137                                    REPEAT A,<0>
1138                                    LIST>
1139
1140                            DEFINE ONES (A),<XLIST
1141                                    REPEAT A,<-1>
1142                                    LIST>
1143
1144                            DEFINE BLOCKR (NAME,EXT,PROT,PROGX,B,F,Z),<
1145                                    NAME
1146                                    EXT
1147                                    EXP PROT*1B8+14B12
1148                                    XWD 1,PROGX
1149                                    EXP Z
1150                                    ZEROES 172
```

```
1151                                    BYTE (4) F (14) 0 (18) B >
1152
1153                            MFDRET: BLOCKR <XWD 1,1>,<SIXBIT /UFD/>,45,1,0,4,20     ;MFD RETREIVAL INFO
1154
1155                            SATBL1: XWD 777776,0            ;THIS SAT BLOCK CONTAINSS
1156                                    ZEROES 5400/44-1        ;ALLOCATION FOR MFD AND SAT BLOCKS.
1157                                    XWD 001777,-1
1158                                    ONES 200*SATBL1-.
1159
1160                            SATRET: BLOCKR <SIXBIT /*SAT*/>,<SIXBIT /SYS/>,555,1,0,1,16     ;SAT RETR. INFO
1161
1162                            MFD1:   XWD 1,1                 ;MFD
1163                                    XWD 654644,0
1164                                    SIXBIT /*SAT*/
1165                                    XWD 637163,17           ;RELOCATED
1166                                    ZEROES 174
1167
1168                            NULBLK: ZEROES 5400/44-1        ;NULL SAT BLOCK
1169                                    XWD 001777,-1
1170                                    ONES 200*NULBLK-.
1171
1172
1173
1174                            >               ;END OF THE REFRESHER FOR THE OLD PDP-6 DISK.
```

```
1175
1176                            IFN     FTRC10, <
1177
1178                            ;DISK REFRESHER FOR THE NEW PDP-10 MODEL RC-10 (BURROUGHS) DISK--
1179
1180                                    EXTERNAL        CPOPJ1,THSDAT,TIME
1181                                    EXTERNAL        MFDBLK,SATXWD,NUMSAT,NUMBLK,SATTOP,SENTSZ
1182                                    EXTERNAL        DISKUP,DFRED,DFWRT,DSKFDG,CHKSUM,RCXCCW,RCXFIN
1183                                    EXTERNAL        K4SWAP
1184                            IFN FTSWAP,<
1185                                    EXTERNAL        MAXSWP,MXK2SWP,BLKSPK
1186                            >
1187                            EXTERNAL LBHIGH
1188                                    INTERNAL        STATBK
1189
1190                    000001  STATBK=1         ;FIXED LOGICAL BLOCK NUMBER FOR THE "VITAL
1191                                             ;  STATISTICS" BLOCK ON THE DISK.
1192                    000170  DSK=170          ;DEFINE DEVICE NUMBER OF THE MODEL RC-10 DISK SYNCHRONIZER.
1193

1194
1195                            ;THESE NEXT TWO PAGES ARE LIFTED DIRECTLY FROM "DSKINT" IN ORDER TO DEFINE THE
1196                            ;  CONO AND CONI BITS USED HERE IN "ONCE".
1197
1198                            ;CONI FORMAT FOR THE MODEL RC-10 DISK SYNCHRONIZER --
1199
1200                            ;BITS 00 - 17   MAINTENANCE PANEL SWITCH SETTING INDICATORS --
1201                                    ;BITS 00 - 04   UNUSED
1202            010000  000000  MSS=1B5          ;SET INDICATES "MAINTENANCE SEGMENT SELECTED" (SEGMENT 81),
1203            004000  000000  PLGSW=1B6        ;1 = WRITE-PROTECT EVERYTHING BELOW (LESS THAN) THE BOUNDARY,
1204                                             ; 0 = WRITE-PROTECT EVERYTHING ABOVE (GREATER THAN) BOUNDARY.
1205                                             ;THE CURRENT IMPLEMENTATION OF THE RC-10 DISK SYNCHRONIZER
1206                                             ;  DOES NOT PERMIT SENSING THE INCLUSION OF THE BOUNDARY.
1207                                    ;BITS 07 - 17   THE WRITE-PROTECTION BOUNDARY (DISK AND TRACK NUMBER).
1208                    400000  DTIP=1B18        ;SET INDICATES DATA TRANSFER IN PROGRESS.
1209                            ;BITS 19 -29    ERROR CONDITION INDICATORS (ERROR WHEN SET) --
1210                    200000          SRCHE=1B19       ;SEARCH ERROR (DISK TIMING TRACK PROBLEMS!!)
1211                    100000          DDF=1B20         ;DISK DESIGNATION ERROR
1212                    040000          TSE=1B21         ;TRACK SELECT ERROR (OR EXCEEDS SYSTEM CAPACITY)
1213                    020000          NRDY=1B22        ;DISK NOT READY (OR NON-EXISTENT DISK REFERENCED)
1214                    010000          PSF=1B23         ;POWER SUPPLY FAILURE
1215                    004000          DPAR=1B24        ;DISK PARITY ERROR
1216                    002000          CHDPAR=1B25      ;CHANNEL DATA PARITY ERROR
1217                    001000          CHCPAR=1B26      ;CHANNEL CONTROL PARITY ERROR
1218                    000400          NXMEM=1B27       ;NON-EXISTENT MEMORY REFERENCED
1219                    000200          WRPE=1B28        ;ATTEMPTED TO WRITE IN PROTECTED DISK AREA (SEE BITS 06 - 17)
1220                    000100          OVR=1B29         ;OVERRUN, I.E,, MEMORY DIDN'T RESPOND QUICKLY ENOUGH
1221                            ;BIT  30        CHANNEL CONTROL WORD WRITTEN IN MEMORY (THIS BIT IS
1222                                             ;TURNED ON ON ALMOST ALL TERMINATIONS,)
1223                    000020  BUSYBT=1B31      ;BUSY (SYNCHRONIZER PERFORMING A COMMAND SEQUENCE)
1224                    000010  DONEFLG=1B32     ;DONE -- THIS ACTUALLY CAUSES THE INTERRUPT,
1225                            ;BITS 33 - 35   PI CHANNEL SELECTION BITS,
1226
1227                            ;COMBINATIONS OF ERROR BITS GROUPED BY TYPE --
```

```
1226                    006000  DATERR=DPAR!CHDPAR              ;DATA ERRORS.
1229                    211100  DEVERR=SRCHE!PSF!CHCPAR!OVR     ;DEVICE ERRORS.
1230                    160600  SFTERR=DDE!TSE!NRDY!WRPE!NXMEM  ;SOFTWARE-PREVENTABLE ERRORS.
```

```
1231                            ;CONO FORMAT FOR THE MODEL RC-10 DISK SYNCHRONIZER --
1232
1233                            ;BITS 00 - 17   UNUSED
1234                            ;BITS 18 - 19   SELECT DISK FOR SECTOR COUNTER READ-OUT (SEE DATAI BITS 28-35)
1235                            ;BITS 20 - 29   RESET THE CORRESPONDING CONI ERROR BIT
1236                                            ;(BUT PSE MAY REFUSE TO BE RESET)
1237                    000040  WRCCWD=1B30     ;WRITE THE CHANNEL CONTROL WORD INTO MEMORY (NOW!)
1238                    000020  STPBIT=1B31     ;STOP -- IMMEDIATELY CEASE PRESENT I/O AND CLEAR THE CHANNEL.
1239                    000010  RESETB=1B32     ;RESET THE DONE FLAG (CORRESPONDING CONI BIT) TO CLEAR INTERRUPT
1240                            ;BITS 33 - 35   PI CHANNEL SELECTION BITS
1241
1242
1243                            ;DATAI FORMAT FOR THE MODEL RC-10 DISK SYNCHRONIZER --
1244
1245                            ;BITS 00 - 17   UNUSED
1246                            ;BITS 18 - 23   PARITY REGISTER
1247                            ;BITS 24 - 25   UNUSED
1248                            ;BITS 26 - 27   DISK SELECTED BY BITS 18-19 OF LAST CONO
1249                            ;BITS 28 - 35   CURRENT SECTOR POSITION OF SELECTED DISK (FOR LATENCY OPTIMIZATION!)

1250
1251
1252                            ;DATAO FORMAT FOR THE MODEL RC-10 DISK SYNCHRONIZER --
1253
1254                            ;BITS 00 - 17   DISK ADDRESS SELECTION
1255                                    ;BITS 00 - 01   DISK SELECT
1256                                    ;BITS 02 - 10   TRACK SELECT (BCD, BUT FIRST CHAR IS JUST 1 BIT)
1257                                    ;BITS 11 - 17   SEGMENT SELECT (BCD, BUT FIRST CHAR HAS ONLY 3 BITS)
1258                            ;BITS 18 - 23   INITIAL PARITY REGISTER SETTING (ZERO EXCEPT FOR DIAGNOSTIC PROGRAMS)
1259                    004000  DDSKPE=1B24     ;DISABLE DISK PARITY ERROR STOP
1260                    002000  DCHNPE=1B25     ;DISABLE CHANNEL DATA PARITY ERROR STOP
1261                    001000  WRBIT=1B26      ;SET MEANS WRITE ON DISK, RESET MEANS READ FROM DISK
1262                            ;BITS 27 - 34   ;CORRESPONDING BITS OF INITIAL CHANNEL CONTROL ADDRESS, HENCE,
1263                                            ; THIS ADDRESS MUST BE EVEN AND IN THE FIRST 1K OF CORE MEMORY!
1264                            ;BIT  35        ;WRITE EVEN PARITY DATA INTO MEMORY (DIAGNOSTICS ONLY!)
```

```
1265
1266
1267  001315                        REFRESH:
1268  001315  260140  000000                PUSHJ   P,DISKUP        ;BE SURE THE DISK IS UP AND READY.
1269  001316  260140  001723'                PUSHJ   P,WRITLK        ;BE SURE WRITE-LOCK SWITCHES ARE OFF.
1270  001317  260140  001660'                PUSHJ   P,CAPCTY        ;DETERMINE THE CURRENT DISK CAPACITY OF THE
1271                                                                  ; SYSTEM, I.E., HOW MANY DISKS ARE ATTACHED TODAY ?
1272  001320  202040  001204'                MOVEM   TAC,LBHIGH      ;STORE HIGHEST LEGAL LOGICAL BLOCK NUMBER.
1273  001321  350000  000001                AOS     TAC
1274                                IFN FTSWAP,<
1275  001322  200100  000531'                MOVE    TAC1,K4SWAP
1276  001323  242100  000000                LSH     TAC1,BLKSPK     ;COMPUTE AND SAVE THE LOWEST LOGICAL BLOCK
1277  001324  276040  000002                SUBM    TAC,TAC1        ; NUMBER RESERVED FOR SWAPPING.
1278  001325  202100  002316'                MOVEM   TAC1,LOWSWP
1279  001326  307100  000001                CAIG    TAC1,STATBK     ;SWAPPING AREA MUST NOT OVERLAY THE
1280  001327  254000  001342'                JRST    REFRSHE         ; VITAL STATISTICS BLOCK.
1281                                >
1282  001330  231040  000000                IDIVI   TAC,NUMBLK
1283  001331  210100  000001                MOVN    TAC1,TAC        ;SET SATXWD TO SHOW ONLY THE NUMBER OF
1284  001332  506100  001252'                HRLM    TAC1,SATXWD     ; SAT BLOCKS ACTUALLY IN EXISTENCE.
1285  001333  550100  000511'                HRRZ    TAC1,MFDBLK
1286  001334  271042  000002                ADDI    TAC,2(TAC1)
1287                                IFN FTSWAP,<
1288  001335  311040  002316'                CAML            TAC,LOWSWP      ;MAKE VARIOUS CONSISTENCY CHECKS TO DETERMINE
1289  001336  254000  001342'                JRST    REFRSHE         ; THAT THE DISK PARAMETERS SPECIFIED DURING THE
1290                                >
1291  001337  307100  000001                CAIG    TAC1,STATBK     ; ONCE-ONLY DIALOGUE DO NOT CAUSE THE DISK
1292  001340  305040  000001                CAIGE   TAC,STATBK      ; SWAPPING AREA, VITAL STATISTICS BLOCK, AND/OR
1293  001341  254000  001361'                JRST    REFRSH1         ; THE MFD AND SAT BLOCKS TO OVERLAP ONE ANOTHER.
1294  001342  201040  001347' REFRSHE: MOVEI   TAC,REFSHME
1295                                                                  ;IF THERE IS SOME OVERLAPPING OF
1296  001343  260140  001002'                PUSHJ   P,ICONM         ; THESE DISK STORAGE AREAS, PRINT AN
1297  001344  260140  000734'                PUSHJ   P,OPOUT         ; ERROR MESSAGE AND THEN GO BACK TO THE
1298  001345  262140  000001                POP     P,TAC           ; DIALOGUE TO ASK THE PARAMETER SPECIFICATION
1299  001346  254000  000475'                JRST    REF6            ; QUESTIONS AGAIN.
1300
1301  001347  502032  240632  REFSHME: ASCIZ   /PARAMETER SPECIFICATION ERROR.  TRY AGAIN.
      001350  426510  551100
      001351  516410  541622
      001352  432230  340650
      001353  446371  620212
      001354  512451  751134
      001355  201012  451262
      001356  202030  740622
1302
1303  001357  471341  505024
1304  001360  050000  000000  /
1305
1306
1307  001361  260140  001564' REFRSH1: PUSHJ   P,WRSTAT        ;WRITE OUT THE VITAL STATISTICS BLOCK ON THE DISK.
1308  001362  254000  002112'                JRST    REFERR          ; AN ERROR ON THIS WRITE IS BAD TROUBLE !!
```

```
1309
1310                                ;NEXT WRITE THE MFD (MASTER FILE DIRECTORY) ONTO THE DISK
1311
1312  001363  260140  002074'          PUSHJ   PDP,WNZCLR        ;CLEAR THE OUTPUT BUFFER
1313  001364  200040  002317'          MOVE    TAC,[XWD 1,1]     ;THE FIRST ENTRY IN THE MFD IS FOR
1314  001365  202040  002116'          MOVEM   TAC,WNZBUF        ; THE MFD ITSELF.
1315  001366  200040  002332'          MOVE    TAC,[SIXBIT/UFD/]
1316  001367  540040  001333'          HRR     TAC,MFDBLK        ;POINTER TO MFD RETRIEVAL INFORMATION
1317  001370  202040  002117'          MOVEM   TAC,WNZBUF+1
1318  001371  200040  002333'          MOVE    TAC,[SIXBIT/*SAT*/]       ;THE ONLY OTHER ENTRY INITIALLY IN THE
1319  001372  202040  002120'          MOVEM   TAC,WNZBUF+2              ; MFD IS FOR A FILE NAMED *SAT*.SYS WHICH
1320  001373  554100  001332'          HLRZ    TAC1,SATXWD               ; PERMITS ACCESS TO THE SAT BLOCKS FOR
1321  001374  550040  001367'          HRRZ    TAC,MFDBLK                ; ANYONE WHO WANTS TO LOOK AT THEM.
1322  001375  275042  777777           SUBI    TAC,-1(TAC1)      ;COMPUTE POINTER TO THE RETRIEVAL INFORMATION
1323  001376  500040  002334'          HLL     TAC,[SIXBIT/SYS/]         ; FOR THE *SAT*.SYS FILE.
1324  001377  202040  002121'          MOVEM   TAC,WNZBUF+3
1325  001400  553000  000001           HRRZS   TAC
1326  001401  271040  000001           ADDI    TAC,1             ;COMPUTE LOGICAL BLOCK NUMBER OF THE MFD.
1327  001402  260140  002033'          PUSHJ   PDP,WNZWR         ;WRITE THE MFD OUT ONTO THE DISK.

1328  001403  254000  002112'          JRST    REFERR            ;ERROR
1329
1330  001404  200040  002335'          MOVE    TAC,[XWD -200,WNZBUF]
1331  001405  260140  000000           PUSHJ   PDP,CHKSUM
1332  001406  552100  002114'          HRRZM   TAC1,CHKSAV       ;SAVE CHECKSUM TO GO INTO RETRIEVAL INFORMATION.
1333
1334
1335                                ;NEXT WRITE MFD RETRIEVAL INFORMATION BLOCK ONTO THE DISK
1336
1337  001407  260140  002074'          PUSHJ   PDP,WNZCLR        ;CLEAR OUTPUT BUFFER
1338  001410  200040  002317'          MOVE    TAC,[XWD 1,1]
1339  001411  202040  002116'          MOVEM   TAC,WNZBUF        ;FILE NAME
1340  001412  550040  000245'          HRRZ    TAC,THSDAT
1341  001413  500040  002332'          HLL     TAC,[SIXBIT /UFD/]
1342  001414  202040  002117'          MOVEM   TAC,WNZBUF+1      ;FILE EXTENSION, DATE
1343  001415  515040  045000           HRLZI   TAC,045000        ;PROTECTION (ANY PROJECT 1 USER MAY WRITE IN MFD)
1344  001416  260140  002100'          PUSHJ   PDP,MDTMOT        ;FILL IN TIME, DATE, AND MODE
1345  001417  202040  002120'          MOVEM   TAC,WNZBUF+2
1346  001420  200040  002336'          MOVE    TAC,[XWD -200,1]          ;NEGATIVE WORD COUNT AND PROGRAMMER NUMBER
1347  001421  202040  002121'          MOVEM   TAC,WNZBUF+3      ;GO INTO WORD 4 OF RETRIEVAL INFO
1348  001422  554100  001373'          HLRZ    TAC1,SATXWD
1349  001423  200040  001374'          MOVE    TAC,MFDBLK
1350  001424  275042  777776           SUBI    TAC,-2(TAC1)      ;COMPUTE POINTER TO THE MFD.
1351  001425  504040  002114'          HRL     TAC,CHKSAV        ;RETRIEVE SAVED CHECKSUM OF MFD.
1352  001426  202040  002122'          MOVEM   TAC,WNZBUF+4
1353  001427  200040  001423'          MOVE    TAC,MFDBLK        ;THE LAST WORD OF ANY BLOCK OF RETRIEVAL INFORMATION
1354  001430  202040  002315'          MOVEM   TAC,WNZBUF+177    ; MUST CONTAIN ITS OWN BLOCK NUMBER.
1355  001431  260140  002033'          PUSHJ   PDP,WNZWR         ;WRITE MFD RETRIEVAL INFORMATION ONTO THE  DISK.
1356  001432  254000  002112'          JRST    REFERR            ;ERROR
```

```
1357
1358                                    ;NEXT CREATE THE NECESSARY SAT BLOCKS ON THE DISK
1359
1360  001433  261140  001422'            PUSH    PDP,SATXWD      ;SATXWD CONTROLS THE OUTER LOOP
1361  001434  201040  001241'            MOVEI   TAC,SATENT      ;SET UP SATPTR TO SHUFFLE THROUGH THE BLOCKS
1362
1363  001435  202040  001304' REFST1:    MOVEM   TAC,SATPTR
1364  001436  260140  002074'            PUSHJ   PDP,WNZCLR      ;CLEAR THE BUFFER
1365  001437  201100  000000            MOVEI   TAC1,NUMBIT
1366  001440  275100  001330'            SUBI    TAC1,NUMBLK
1367  001441  201040  001440'            MOVEI   TAC,NUMBLK
1368  001442  260140  002061'            PUSHJ   PDP,SETWNZ      ;MARK UNAVAILABLE BITS AT END OF THE BLOCK.
1369                                    IFN FTSWAP,<
1370  001443  554120  001435'            HLRZ    TAC1,@SATPTR    ;GET FIRST LOGICAL BLOCK NUMBER IN THIS SAT BLOCK
1371  001444  315100  002316'            CAMGE   TAC1,LOWSWP     ;IS ENTIRE BLOCK WITHIN SWAPPING AREA?
1372  001445  254000  001452'            JRST    REFST2          ;NO, JUMP AHEAD
1373  001446  201100  001437'            MOVEI   TAC1,NUMBIT     ;YES, MARK IT ALL OFF LIMITS
1374  001447  201040  000000            MOVEI   TAC,0
1375  001450  260140  002061'            PUSHJ   PDP,SETWNZ
1376  001451  254000  001510'            JRST    REFST7          ;HAVING SET ALL BITS, NO FURTHER TESTING IS NEEDED.
1377
1378  001452  210040  000002  REFST2:    MOVN    TAC,TAC1
1379  001453  271100  001441'            ADDI    TAC1,NUMBLK
1380  001454  274100  002316'            SUB     TAC1,LOWSWP     ;IS ANY OF SWAPPING AREA WITHIN THIS SAT BLOCK?
1381  001455  323100  001460'            JUMPLE  TAC1,REFST3     ;NO, JUMP AHEAD
1382  001456  270040  002316'            ADD     TAC,LOWSWP      ;YES, MARK SWAPPING AREA AS UNAVAILABLE.
1383  001457  260140  002061'            PUSHJ   PDP,SETWNZ
1384                                    >
1385  001460  554120  001443' REFST3:    HLRZ    TAC1,@SATPTR    ;GET FIRST LOGICAL BLOCK NUMBER IN THIS SAT BLOCK.
1386  001461  326100  001464'            JUMPN   TAC1,RFST3A     ;ONLY FOR FIRST SAT BLOCK--
1387  001462  515040  400000            HRLZI   TAC,400000      ;MARK LOGICAL BLOCK 0 AS BEING IN USE.
1388  001463  436040  002116'            IORM    TAC,WNZBUF
1389  001464  551040  000001  RFST3A:    HRRZI   TAC,STATBK
1390  001465  301042  000000            CAIL    TAC,(TAC1)      ;IS THE VITAL STATISTICS BLOCK WITHIN
1391  001466  301042  001453'            CAIL    TAC,NUMBLK(TAC1)        ; THE RANGE OF THIS SAT BLOCK ?
1392  001467  254000  001473'            JRST    REFST4          ;NO, JUMP AHEAD
1393  001470  274040  000002            SUB     TAC,TAC1
1394  001471  201100  000001            MOVEI   TAC1,1          ;YES, MARK IT UNAVAILABLE.
1395  001472  260140  002061'            PUSHJ   PDP,SETWNZ
```

```
1396
1397  001473  200040  001433' REFST4: MOVE    TAC,SATXWD      ;SET UP TO LOOP THRU LOGICAL BLOCK NUMBERS FOR MFD,
1398  001474  274040  002337'          SUB     TAC,[XWD 3,1]   ; MFD RETRVL INFO, SAT BLOCKS, AND SAT RETRVL INFO,
1399
1400  001475  261140  000001  REFST5: PUSH    PDP,TAC
1401  001476  553000  000001          HRRZS   TAC
1402  001477  554120  001460'          HLRZ    TAC1,@SATPTR
1403  001500  301042  000000          CAIL    TAC,(TAC1)      ;IS THE INDICATED LOGICAL BLOCK NUMBER WITHIN
1404  001501  301042  001466'          CAIL    TAC,NUMBLK(TAC1)        ; THIS PARTICULAR SAT BLOCK ?
1405  001502  254000  001506'          JRST    REFST6          ;NO, JUMP AHEAD
1406  001503  274040  000002          SUB     TAC,TAC1
1407  001504  201100  000001          MOVEI   TAC1,1          ;YES, MARK IN-USE BIT,
1408  001505  260140  002061'          PUSHJ   PDP,SETWNZ
1409
1410  001506  262140  000001  REFST6: POP     PDP,TAC         ;LOOP THROUGH THE WHOLE SET OF SPECIAL BLOCKS
1411  001507  253040  001475'          AOBJN   TAC,REFST5      ; WHICH MUST BE MARKED UNAVAILABLE,
1412
1413
1414  001510  200040  002335' REFST7: MOVE    TAC,[XWD -200,WNZBUF]

1415  001511  260140  001405'          PUSHJ   PDP,CHKSUM      ;COMPUTE CHECKSUM
1416  001512  200040  001477'          MOVE    TAC,SATPTR
1417  001513  552121  000001          HRRZM   TAC1,1(TAC)     ;STORE IT TEMPORARILY IN THE SATENT TABLE,
1418
1419  001514  550043  000000          HRRZ    TAC,(PDP)       ;LOGICAL BLOCK NUMBER TO BE WRITTEN
1420  001515  260140  002033'          PUSHJ   PDP,WNZWR       ;WRITE THIS SAT BLOCK ONTO THE DISK
1421  001516  254000  002111'          JRST    REFER1          ;ERROR
1422
1423  001517  262140  000001          POP     PDP,TAC
1424  001520  253040  001522'          AOBJN   TAC,.+2         ;LAST SAT BLOCK WRITTEN ?
1425
1426  001521  254000  001526'          JRST    REFSRF          ;YES, GO WRITE SAT RETRIEVAL INFORMATION,
1427
1428  001522  261140  000001          PUSH    PDP,TAC         ;NO, ADVANCE SATPTR AND LOOP
1429  001523  200040  001512'          MOVE    TAC,SATPTR      ; BACK TO DO NEXT SAT BLOCK,
1430  001524  271040  001307'          ADDI    TAC,SENTSZ
1431  001525  254000  001435'          JRST    REFST1
```

```
1432
1433                                ;FINALLY, WRITE OUT THE RETRIEVAL INFORMATION FOR THE FILE *SAT*.SYS
1434
1435  001526  260140  002074' REFSRF: PUSHJ   PDP,WNZCLR       ;CLEAR THE BUFFER
1436  001527  200040  002333'         MOVE    TAC,[SIXBIT /*SAT*/]
1437  001530  202040  002116'         MOVEM   TAC,WNZBUF       ;STORE FILE NAME
1438  001531  550040  001412'         HRRZ    TAC,THSDAT
1439  001532  500040  002334'         HLL     TAC,[SIXBIT /SYS/]
1440  001533  202040  002117'         MOVEM   TAC,WNZBUF+1     ;STORE EXTENSION AND DATE
1441  001534  515040  555000          HRLZI   TAC,555000       ;PROTECTION CODE (UNTOUCHABLE FILE ! )
1442  001535  260140  002100'         PUSHJ   PDP,MDTMDT       ;FILL IN TIME, DATE, AND MODE
1443  001536  202040  002120'         MOVEM   TAC,WNZBUF+2
1444  001537  510040  001473'         HLLZ    TAC,SATXWD       ;COMPUTE NEGATIVE WORD COUNT
1445  001540  242040  000007          LSH     TAC,7
1446  001541  541040  000001          HRRI    TAC,1            ;PROGRAMMER NUMBER=1
1447  001542  202040  002121'         MOVEM   TAC,WNZBUF+3
1448
1449
1450  001543  201040  001434'         MOVEI   TAC,SATENT       ;PREPARE TO STORE POINTERS TO ALL THE SAT BLOCKS,
1451  001544  200100  001537'         MOVE    TAC1,SATXWD
1452  001545  201000  000000          MOVEI   AC2,0
1453
1454  001546  550640  000032 REFSRL: HRRZ    AC1,TAC1         ;GET LOGICAL BLOCK NUMBER OF SAT BLOCK,
1455  001547  504641  000001          HRL     AC1,1(TAC)       ;RETRIEVE THE CORRESPONDING CHECK-SUM,
1456  001550  202656  002122'         MOVEM   AC1,WNZBUF+4(AC2)        ;STORE A POINTER WORD IN THE BUFFER,
1457  001551  205640  400000          MOVSI   AC1,400000       ;RESTORE SATENT ENTRY TO CORRECT INITIAL VALUE,
1458  001552  202641  000001          MOVEM   AC1,1(TAC)
1459  001553  350000  000016          AOS     AC2
1460  001554  271040  001524'         ADDI    TAC,SENTSZ
1461  001555  253100  001546'         AOBJN   TAC1,REFSRL      ;IS THAT ALL THE POINTERS ?   IF NOT, LOOP BACK,
1462
1463
1464  001556  200040  001427'         MOVE    TAC,MFDBLK       ;YES,
1465  001557  271056  000001          ADDI    TAC,1(AC2)       ;NOW COMPUTE LOGICAL BLOCK NUMBER OF THIS SAT
1466  001560  202040  002315'         MOVEM   TAC,WNZBUF+177           ; RETRIEVAL INFORMATION, AND STORE IT IN
1467                                                           ; THE LAST WORD OF THE BLOCK (THIS IS
1468                                                           ; REQUIRED FORMAT OF ALL RETRVL INFO),
1469
1470  001561  260140  002033'         PUSHJ   PDP,WNZWR        ;WRITE THE *SAT*.SYS RETRVL INFO ONTO THE DISK,
1471  001562  254000  002112'         JRST    REFERR           ;ERROR
1472
1473
1474
1475  001563  254000  000715'         JRST    CPOPJ1           ;***EXIT FROM PDP-10 DISK REFRESHER.***.......
```

```
1476
1477                            ;ROUTINE TO WRITE THE FIXED BLOCK OF VITAL STATISTICS ONTO THE DISK.
1478
1479
1480  001564  260140  002074' WRSTAT: PUSHJ   P,WNZCLR              ;CLEAR THE OUTPUT BUFFER
1481  001565  200040  002340'         MOVE    TAC,[XWD -WRSTSZ,0]
1482  001566  200121  001617' WRSTA1: MOVE    TAC1,@WRSTA9(TAC)      ;STORE KEY QUANTITIES IN THE OUTPUT BUFFER
1483  001567  202101  002116'         MOVEM   TAC1,WNZBUF(TAC)
1484  001570  253040  001566'         AOBJN   TAC,WRSTA1
1485
1486  001571  200040  002341'         MOVE    TAC,[SIXBIT/STATBK/]   ;STORE AN IDENTIFIER NEAR THE
1487  001572  202040  002314'         MOVEM   TAC,WNZBUF++D126       ; END OF THE VITAL STATISTICS BLOCK.
1488
1489  001573  201040  000001         MOVEI   TAC,STATBK
1490  001574  260140  002033'         PUSHJ   P,WNZWR        ;WRITE THIS VITAL STATISTICS BLOCK ONTO THE DISK.
1491  001575  254000  001577'         JRST    WRSTA2         ;WRITE ERROR
1492  001576  254000  001563'         JRST    CPOPJ1         ;GOOD WRITE,   ***SUBROUTINE EXIT***.....
1493
1494

1495  001577  201040  001603' WRSTA2: MOVEI   TAC,WRSTM1
1496  001600  260140  001002'         PUSHJ   P,ICONM        ;PRINT ERROR MESSAGE IF THIS VITAL WRITE FAILS,
1497  001601  260140  000734'         PUSHJ   P,OPOUT
1498  001602  263140  000000         POPJ    P,             ;THEN ALSO GIVE AN ERROR RETURN FOR FURTHER ACTION,
1499
1500  001603  416031  647236  WRSTM1: ASCIZ   /CANNOT WRITE VITAL STATISTICS BLOCK ON DISK.   HELP!!
      001604  521012  751222
      001605  522124  053222
      001606  522031  420246
      001607  522032  444646
      001610  522230  351500
      001611  412311  741626
      001612  202371  620210
      001613  446471  327100
      001614  201011  042630
1501  001615  501024  106424  /
      001616  000000  000000
1502
1503
1504                            ;TABLE OF POINTERS TO THE VITAL STATISTICS THAT ARE STORED IN "STATBK" ON THE DISK,
1505
1506  001617  000000  001556' WRSTA9: XWD     ,MFDBLK        ;LOGICAL BLOCK NUMBER OF MFD RETRIEVAL INFORMATION
1507  001620  000000  001544'         XWD     ,SATXWD        ;LH = - NUMBER OF SAT BLOCKS ACTUALLY IN EXISTENCE,
1508                                                         ;RH = LOGICAL BLOCK NUMBER OF FIRST SAT BLOCK
1509  001621  000000  001320'         XWD     ,LBHIGH        ;HIGHEST LEGAL LOGICAL BLOCK NUMBER ON DISK(S)
1510  001622  000000  002316'         XWD     ,LOWSWP        ;LOWEST LOGICAL BLOCK NUMBER ALLOCATED FOR SWAPPING
1511  001623  000000  001322'         XWD     ,K4SWAP        ;NUMBER OF 1K BLOCKS OF DISK SPACE FOR SWAPPING
1512                            IFN FTSWAP,<
1513  001624  000000  000000         XWD     ,MAXSWP        ;MAXIMUM NUMBER OF THOSE 1K BLOCKS ACTUALLY
1514                                                         ; USED SINCE THE DISK WAS LAST REFRESHED.
1515                            >
1516                            IFE FTSWAP,<
1517                                    XWD     0,LOWSWP       ;LOWSWP DOES DOUBLE DUMMY DUTY IN
1518                                                           ; NON-SWAPPING SYSTEM
```

```
1519                            >
1520
1521                    000006  WRSTSZ=.-WRSTA9
```

```
1522
1523                            ;ROUTINE TO READ IN THE FIXED BLOCK OF "VITAL STATISTICS" FROM
1524                            ;  THE DISK AND DISTRIBUTE THEM TO THE APPROPRIATE CORE LOCATIONS.
1525
1526
1527  001625  201040  000001  RDSTAT: MOVEI   TAC,STATBK
1528  001626  260140  002013'          PUSHJ   PDP,WNZRD       ;READ IN THE VITAL STATISTICS BLOCK.
1529  001627  254000  001640'          JRST    RDSTA2          ;ERROR
1530  001630  200040  002314'          MOVE    TAC,WNZBUF+.D126                ;CHECK THE IDENTIFIER TO VERIFY THAT
1531  001631  312040  002341'          CAME    TAC,[SIXBIT     /STATBK/]       ; THE CORRECT BLOCK WAS READ.
1532  001632  254000  001640'          JRST    RDSTA2          ;ERROR, STATISTICS BLOCK APPARENTLY WAS CLOBBERED.
1533
1534  001633  200040  002340'          MOVE    TAC,[XWD -WRSTSZ,0]     ;STORE AWAY THESE IMPORTANT PARAMETERS
1535  001634  200101  002116' RDSTA1: MOVE    TAC1,WNZBUF(TAC)        ; UNDER CONTROL OF A TABLE IN THE "WRSTAT"
1536  001635  202121  001617'          MOVEM   TAC1,@WRSTA9(TAC)       ; ROUTINE WHICH WRITES OUT THIS VITAL
1537  001636  253040  001634'          AOBJN   TAC,RDSTA1              ; STATISTICS BLOCK ONTO THE DISK.
1538
1539  001637  263140  000000          POPJ    PDP,            ;***SUBROUTINE EXIT***.......
1540

1541
1542  001640  201040  001645' RDSTA2: MOVEI   TAC,RDSTM1
1543                                                          ;PRINT ERROR MESSAGE IF THE VITAL
1544  001641  260140  001002'          PUSHJ   PDP,ICONM       ; STATISTICS BLOCK CANNOT BE READ
1545  001642  260140  000734'          PUSHJ   PDP,OPOUT
1546  001643  260140  000467'          PUSHJ   PDP,REF         ;GO BACK THRU THE REFRESH DIALOGUE IN "ONCE",
1547  001644  254000  001625'          JRST    RDSTAT          ;NOW TRY AGAIN.
1548
1549  001645  532232  440630  RDSTM1: ASCIZ   /VITAL DISK STATISTICS LOST, SUGGEST REFRESHING!
      001646  202111  151626
      001647  202472  440650
      001650  446472  444606
      001651  515011  447646
      001652  521344  051652
      001653  436170  551650
      001654  202450  543244
      001655  426471  044634
1550  001656  435021  505032
1551
1552  001657  050240  000000  /
```

```
1553
1554                            ;ROUTINE TO DETERMINE THE CURRENT SYSTEM'S DISK CAPACITY, I.E., THE NUMBER OF
1555                            ;  DISKS ATTACHED TO THE RC-10 DISK SYNCHRONIZER TODAY.
1556
1557                            ;ON EXIT FROM THIS SUBROUTINE,
1558                            ;                   C(TAC) = HIGHEST LOGICAL DISK BLOCK NUMBER NOW EXTANT.
1559
1560  001660  201040  000000  CAPCTY: MOVEI   TAC,NUMSAT
1561  001661  221040  001501'         IMULI   TAC,NUMBLK      ;COMPUTE AND SAVE MAXIMUM POSSIBLE CAPACITY.
1562  001662  261140  001621'         PUSH    PDP,LBHIGH      ;SAVE EXISTING VALUE OF HIGHEST LEGAL BLOCK NUMBER,
1563  001663  202040  001662'         MOVEM   TAC,LBHIGH      ;TEMPORARILY CHANGE THIS VALUE SO THAT THE VALIDITY
1564                                                          ; CHECK IN "DSKINT" WILL NOT FAIL SPURIOUSLY.
1565
1566  001664  201040  000000         MOVEI   TAC,0
1567  001665  261140  000001  CPCT1:  PUSH    PDP,TAC
1568  001666  260140  002213'         PUSHJ   PDP,UNZRO       ;ATTEMPT TO READ FIRST BLOCK OF EACH SUCCESSIVE DISK,
1569  001667  254000  001677'         JRST    CPCT3           ;ERROR
1570  001670  262140  000001         POP     PDP,TAC
1571  001671  271040  001661'         ADDI    TAC,NUMBLK
1572  001672  315040  001663'         CAMGE   TAC,LBHIGH      ;ALREADY TRIED ALL DISKS (MAXIMUM CAPACITY) ?
1573  001673  254000  001665'         JRST    CPCT1           ;NO, GO TRY THE NEXT ONE.
1574
1575  001674  275040  000001  CPCT2:  SUBI    TAC,1
1576  001675  262140  001672'         POP     PDP,LBHIGH      ;RESTORE PREVIOUSLY EXISTING VALUE OF THE HIGHEST
1577                                                          ; LEGAL LOGICAL DISK BLOCK NUMBER.
1578  001676  263140  000000         POPJ    PDP,            ;***SUBROUTINE EXIT***.......
1579
1580
1581  001677  262140  000001  CPCT3:  POP     PDP,TAC
1582  001700  717300  020000         CONSZ   DSK,NRDY        ;NON-EXISTENT DISK?
1583  001701  326040  001674'         JUMPN   TAC,CPCT2       ;YES, THE LIMIT OF DISK STORAGE HAS BEEN FOUND,
1584                                                          ; HOWEVER, DISK 0 MUST ALWAYS BE PRESENT.
1585
1586  001702  201040  001707'         MOVEI   TAC,CPCTM1      ;NO, SOME OTHER TYPE OF READ ERROR,
1587  001703  260140  001002'         PUSHJ   PDP,ICONM
1588  001704  260140  000734'         PUSHJ   PDP,OPOUT       ;PRINT ERROR MESSAGE
1589  001705  262140  001675'         POP     PDP,LBHIGH
1590  001706  254000  001660'         JRST    CAPCTY          ;TRY AGAIN
1591
1592  001707  422232  345500  CPCTM1: ASCIZ   /DISK READ ERROR DURING CAPACITY SEARCH INITIALIZATION
      001710  512130  142100
      001711  426452  247644
      001712  202112  551222
      001713  472164  041672
      001714  502030  344650
      001715  545012  342602
      001716  512071  020222
      001717  472232  444602
      001720  462233  240650
1593  001721  446371  606424
1594  001722  050000  000000  /
```

```
1595
1596                                    ;ROUTINE TO ASSURE THAT THE STATUS OF THE WRITE-LOCK SWITCHES ON THE
1597                                    ;  DISK MAINTENANCE PANEL IS SUCH THAT THEY ARE PROTECTING "LESS THAN 0020".
1598
1599                                    ;THIS ROUTINE LOOPS PRINTING AN ERROR MESSAGE UNTIL THE SWITCHES ARE O.K.
1600
1601  001723  717240  000001  WRITLK: CONI    DSK,TAC
1602  001724  607040  013777          TLNN    TAC,13777          ;BOUNDARY ZERO AND MAINTENANCE-SEGMENT OFF ?
1603  001725  607040  004000          TLNN    TAC,<PLGSW>B53     ;YES, IS ROTARY SWITCH SET
1604                                                             ; TO EITHER X UNDER OR OVER(AS OPPOSED
1605                                                             ; TO UNDER OR X OVER)
1606  001726  254000  001741'          JRST    WRLK8              ;NO, GO PRINT ERROR MESSAGE AND LOOP.
1607
1608                                    ;AN UNFORTUNATE HARDWARE OVERSIGHT MAKES IT IMPOSSIBLE TO SENSE THE DIFFERENCE
1609                                    ;  BETWEEN "PROTECT LESS THAN" AND "PROTECT LESS THAN OR EQUAL".  THEREFORE,
1610                                    ;  THIS ROUTINE MUST ACTUALLY TRY WRITING ON TRACK 0000.
1611  001727  200040  002342'          MOVE    TAC,[252525252525]
1612  001730  202040  002116'          MOVEM   TAC,WNZBUF
1613  001731  200040  002343'          MOVE    TAC,[XWD WNZBUF,WNZBUF+1]

1614  001732  251040  002315'          BLT     TAC,WNZBUF+177
1615  001733  201040  000000          MOVEI   TAC,0
1616  001734  260140  002033'          PUSHJ   PDP,WNZWR          ;WRITE PATTERN IN BLOCK 0.
1617  001735  254000  001737'          JRST    WRLK2              ;ERROR
1618  001736  263140  000000          POPJ    PDP,               ;***GOOD EXIT FROM WRITLK SUBROUTINE***,.....
1619
1620
1621  001737  201040  001745' WRLK2:  MOVEI   TAC,WRLKM1
1622  001740  717300  200200          CONSZ   DSK,WRPE           ;WRITE-PROTECTION ERROR ?
1623                                                             ;YES, PROTECT SWITCH ERRONEOUSLY SET ON "LESS THAN
1624                                                             ; OR EQUAL" INSTEAD OF "LESS THAN".
1625                                                             ;NO, SOME OTHER FAILURE DURING THE WRITE OF BLOCK 0.
1626  001741  201040  001760' WRLK8:  MOVEI   TAC,WRLKM2        ;WRITE-LOCK SWITCHES SET INCORRECTLY.
1627  001742  260140  001002' WRLK9:  PUSHJ   PDP,ICONM         ;PRINT ERROR MESSAGE
1628  001743  260140  000734'          PUSHJ   PDP,OPOUT
1629  001744  254000  001723'          JRST    WRITLK             ;GO TRY AGAIN
1630
1631  001745  422272  345500  WRLKM1: ASCIZ   /DISK BLOCK 0 WRITE ERROR DURING WRITE-LOCK CHECK.
      001746  412311  741626
      001747  201474  053644
      001750  446510  523212
      001751  512451  751100
      001752  422532  244634
      001753  435212  751222
      001754  522125  546236
      001755  415264  041620
      001756  426071  327032
1632
1633  001757  050240  000000  /
1634  001760  212330  144634  WRLKM2: ASCIZ   /"MAINTENANCE SEGMENT" SWITCH MUST BE OFF, AND
      001761  522131  647634
      001762  416124  051612
      001763  436330  547250
      001764  211012  353622
```

```
        001765  522071  020232
        001766  526472  420204
        001767  425011  743214
        001770  261010  147210
 1635   001771  064250  444646  DISK WRITE PROTECTION SWITCHES MUST BE SET TO 0000 AND "LESS THAN".
        001772  455012  751222
        001773  522124  050244
        001774  476510  541650
        001775  446371  620246
        001776  536232  441620
        001777  426464  046652
        002000  516504  041212
        002001  202470  552100
        002002  522364  030140
        002003  301404  040634
        002004  421004  246212
        002005  516464  052220
        002006  406344  227032
 1636   002007  052151  154120  FIX THEM ! !
        002010  522210  546500
 1637   002011  205004  106424
 1638   002012  050000  000000  /
```

```
1639
1640                            ;DISK READING ROUTINE FOR USE ONLY DURING ONCE-ONLY CODE,
1641                            ;  AVOIDS USING THE MONITOR QUEUES AND INTERRUPT STRUCTURE.
1642
1643                            ;ON ENTRY TO THIS SUBROUTINE, TAC CONTAINS THE LOGICAL BLOCK NUMBER OF THE DISK
1644                            ;  BLOCK WHICH IS TO BE READ INTO THE ONCE-ONLY DISK BUFFER AREA.
1645
1646  002013  717200  377710  WNZRD:  CONO    DSK,DATERR!DEVERR!SETERR!RESETB         ;CLEAR THE DISK SYNCHRONIZER
1647  002014  717000  002030          CONSZ   DSK,BUSYBT!DONEFLG
1648  002015  254000  002013'          JRST    WNZRD                   ;DON'T PROCEED UNLESS THE CLEAR WORKED.
1649
1650  002016  261140  000000          PUSH    PDP,DSKFDG      ;SAVE THIS INSTRUCTION FROM "DSKINT",
1651  002017  200100  002344'          MOVE    TAC1,[HRRZI TAC,0]      ;NOW TAMPER WITH THAT SAVED LOCATION SO
1652  002020  202100  002016'          MOVEM   TAC1,DSKFDG     ; THAT DSKINT WILL NOT ENABLE ANY INTERRUPT CHANNEL.
1653  002021  200100  002345'          MOVE    TAC1,[XWD -200,WNZBUF-1]
1654  002022  260140  000000          PUSHJ   PDP,DFRED       ;CALL DEVICE-DEPENDENT ROUTINE IN "DSKINT" DIRECTLY.
1655  002023  262140  002020' WNZRDC: POP     PDP,DSKFDG      ;RESTORE THE MOLESTED INSTRUCTION IN DSKINT.
1656
1657  002024  260140  002045'          PUSHJ   PDP,WNZWAT      ;WAIT UNTIL THE READ IS FINISHED.

1658
1659  002025  717200  377700          CONSZ   DSK,DATERR!DEVERR!SETERR        ;ANY ERRORS ?
1660  002026  254000  002032'          JRST    WNZRDX          ;YES, GIVE ERROR RETURN.
1661  002027  201100  000000          MOVE    TAC1,RDXCCW     ;NO, BUT CHECK THAT THE FINAL VALUE OF THE
1662  002030  316100  002040          CAME    TAC1,RDXFIN     ; CHANNEL CONTROL WORD WAS AS EXPECTED ALSO.
1663  002031  330473  002030          AOS     (PDP)           ;***GOOD RETURN***.....
1664  002032  263140  000000  WNZRDX: POPJ    PDP,            ;ERROR RETURN,
1665
1666
1667
1668
1669
1670
1671                            ;DISK WRITING ROUTINE FOR USE ONLY DURING ONCE-ONLY CODE,
1672                            ;  AVOIDS USING THE MONITOR QUEUES AND INTERRUPT STRUCTURE.
1673
1674                            ;ON ENTRY TO THIS SUBROUTINE, TAC CONTAINS THE LOGICAL BLOCK NUMBER OF THE DISK
1675                            ;  BLOCK WHICH IS TO BE WRITTEN ONTO THE DISK FROM THE ONCE-ONLY DISK BUFFER AREA.
1676
1677  002033  717200  377710  WNZWR:  CONO    DSK,DATERR!DEVERR!SETERR!RESETB         ;CLEAR THE DISK SYNCHRONIZER.
1678  002034  717000  002030          CONSZ   DSK,BUSYBT!DONEFLG
1679  002035  254000  002033'          JRST    WNZWR                   ;DON'T PROCEED UNLESS THE CLEAR WORKED.
1680
1681  002036  202040  002115'          MOVEM   TAC,LBSAVE      ;SAVE LOGICAL BLOCK NUMBER IN CASE AN ERROR RETURN
1682                                                            ; MUST BE TAKEN FROM THE "REFRESH" SUBROUTINE.
1683  002037  261140  002023'          PUSH    PDP,DSKFDG      ;SAVE THIS INSTRUCTION FROM "DSKINT",
1684  002040  200100  002344'          MOVE    TAC1,[HRRZI TAC,0]      ;NOW TAMPER WITH THAT SAVED LOCATION SO
1685  002041  202100  002037'          MOVEM   TAC1,DSKFDG     ; THAT DSKINT WILL NOT ENABLE ANY INTERRUPT CHANNEL.
1686  002042  200100  002345'          MOVE    TAC1,[XWD -200,WNZBUF-1]
1687  002043  260140  000000          PUSHJ   PDP,DFWRT       ;CALL DEVICE-DEPENDENT ROUTINE IN "DSKINT" DIRECTLY.
1688
1689  002044  254000  002023'          JRST    WNZRDC          ;THE REMAINDER OF THE WRITE ROUTINE IS IDENTICAL
1690                                                            ; TO THE READ ROUTINE.
1691
```

```
1692                            ;SUBROUTINE TO WATCH FOR THE DONE FLAG TO COME ON AND TIME OUT IF
1693                            ; IT DOESN'T COME UP SOON ENOUGH (SAY 5 SEC OR SO).
1694
1695  002045  200040  002346' WNZWAT: MOVE    TAC,[-D1000000] ;AT MOST ONE MILLION TIMES THRU THE CONSZ+SOJG LOOP.
1696  002046  717300  000010  WNZWTL: CONSZ   DSK,DONEFLG     ;DONE FLAG ON YET ?
1697  002047  263140  000000          POPJ    P,              ;YES, GOOD EXIT.....
1698  002050  367040  002046'         SOJG    TAC,WNZWTL      ;NO, TIMED OUT YET ?
1699                                                          ;YES, PRINT "DISK HUNG" ERROR MESSAGE.
1700  002051  201040  002056'         MOVEI   TAC,WATMSG
1701  002052  260140  001002'         PUSHJ   P,ICONM
1702  002053  260140  000734'         PUSHJ   P,OPOUT
1703  002054  262140  000001          POP     P,TAC           ;SKIP UP A SUBROUTINE LEVEL, AND GIVE AN
1704  002055  263140  000000          POPJ    P,              ; ERROR RETURN FROM WNZRD OR WNZWR.....
1705  002056  422232  345500  WATMSG: ASCIZ   /DISK HUNG.
      002057  442531  643534
1706  002060  264240  000000  /
1707
1708
1709                            ;SUBROUTINE TO SET SPECIFIED BITS IN A SAT BLOCK WHILE FORMING IT IN WNZBUF
1710
1711                            ;ENTRY CONDITIONS --    C(TAC) = FIRST BIT NUMBER TO SET (NUMBERED BEGINNING AT 0)
1712                            ;                       C(TAC1) = NUMBER OF BITS TO SET
1713
1714  002061  200640  000002  SETWNZ: MOVE    AC1,TAC1
1715  002062  231040  000044          IDIVI   TAC,^D36
1716  002063  205200  400000          MOVSI   AC2,400000
1717  002064  213040  000002          MOVNS   TAC1
1718  002065  241202  000000          ROT     AC2,(TAC1)
1719  002066  436201  002116' STWNZ1: ORM     AC2,WNZBUF(TAC)
1720  002067  241200  777777          ROT     AC2,-1
1721  002070  327200  002072'         JUMPG   AC2,STWNZ2
1722  002071  271040  000001          ADDI    TAC,1
1723  002072  366640  002066' STWNZ2: SOJN    AC1,STWNZ1
1724  002073  263140  000000          POPJ    P,              ;SUBROUTINE EXIT.....
1725
1726
1727                            ;SUBROUTINE TO CLEAR THE ONCE-ONLY OUTPUT BUFFER, WNZBUF
1728  002074  402020  002116' WNZCLR: SETZM   WNZBUF
1729  002075  200040  002343'         MOVE    TAC,[XWD WNZBUF,WNZBUF+1]
1730  002076  251040  002315'         BLT     TAC,WNZBUF+177
1731  002077  263140  000000          POPJ    P,              ;SUBROUTINE EXIT.....
```

```
1732
1733                            ;SUBROUTINE TO MASK THE TIME, DATE, AND DATA MODE (=14 FOR BINARY MODE) INTO ACCUMU-
1734                            ;  LATOR TAC IN THE FORMAT USED IN WORD 3 OF A DISK RETRIEVAL INFORMATION BLOCK.
1735
1736  002100  261140  000001  MDTMDT: PUSH    P,TAC
1737  002101  200040  001000'         MOVE    TAC,TIME        ;GET TIME IN JIFFIES.
1738  002102  231040  000274'         IDIVI   TAC,JIFMIN      ;CONVERT TO MINUTES
1739  002103  242040  000014         LSH     TAC,↑D12
1740  002104  434040  001531'         IOR     TAC,THSDAT      ;TODAY'S DATE
1741  002105  661040  000600         TLO     TAC,14B<↑D12+↑D18>      ;BINARY MODE
1742  002106  436043  000000         IORM    TAC,(P)
1743  002107  262140  000001         POP     P,TAC
1744  002110  263140  000000         POPJ    P,              ;SUBROUTINE EXIT.........
1745
1746
1747
1748                            ;ERROR EXIT FROM "REFRESH" WHEN A BAD WRITE ON THE DISK OCCURS DURING REFRESHING.
1749
1750  002111  262140  000001  REFER1: POP     P,TAC           ;CORRECT P IF EXTRA PUSH HAD OCCURRED.

1751  002112  200040  002115' REFERR: MOVE    TAC,LBSAVE      ;GET BLOCK NUMBER FOR THE ERROR MESSAGE AT REF4.
1752  002113  263140  000000         POPJ    P,              ;THIS NO-SKIP RETURN RE-ENTERS THE REFRESH DIALOGUE.
1753
1754
1755
1756                            ;MISCELLANEOUS STORAGE SPACE USED BY THE PDP-10 DISK REFRESHING ROUTINE.
1757
1758  002114  000000  000000  CHKSAV: 0
1759  002115  000000  000000  LBSAVE: 0
1760
1761  002116                  WNZBUF: BLOCK   200     ;BUFFER FOR READING AND WRITING DISK
1762                                                  ; BLOCKS DURING THE ONCE-ONLY CODE.
1763
1764  002316  000000  000000  LOWSWP: 0               ;LOWEST LOGICAL BLOCK FOR SWAPPING
1765
1766                                  >               ;END OF REFRESH PROCEDURE FOR THE RD10 DISK.
1767
1768                                  >               ;CLOSE AN EVEN EARLIER FTDISK CONDITIONAL.
```

```
1769                            INTERNAL FTCHECK
1770                            IFN FTCHEC,<
1771
1772                            EXTERNAL CHKBEG,CHKEND,CHECK,MONPTR,MONSUM
1773
1774                            CK:     MOVEI TAC,CHKBEG
1775                                    SUBI TAC,CHKEND
1776                                    HRLI TAC,CHKBEG
1777                                    MOVSS TAC
1778                                    PUSHJ PDP,CHECK         ;COMPUTE AND STORE CHECKSUM OF THE PURE AREA OF
1779                                    MOVEM TAC1,MONSUM       ; THE TIME-SHARING MONITOR.
1780                                    POPJ PDP,
1781
1782                            >
1783
1784  002317                    ONCEND: END
1785  002317  000001  000001
1786  002320  000067' 000040
1787  002321  264000  000125'
1788  002322  376776  000400
1789  002323  777760  001026'
1790  002324  200000  000000
1791  002325  331007  000052'
1792  002326  440600  000004
1793  002327  236460  000000
1794  002330  436471  000000
1795  002331  000011  111111
1796  002332  654644  000000
1797  002333  126341  641200
1798  002334  637163  000000
1799  002335  777600  002116'
1800  002336  777600  000001
1801  002337  000003  000001
1802  002340  777772  000000
1803  002341  636441  644253
1804  002342  252525  252525
1805  002343  002116' 002117'
1806  002344  551040  000000
1807  002345  777600  002115'
1808  002346  000003  641100
```

NO ERRORS DETECTED

PROGRAM BREAK IS 002347

SYMBOL TABLE

AC1		000015	INT	AC2		000016	INT	ALRCHN		000060'	
ALRLNK		000340'		AND2RG		001132'	EXT	APRCHK		000775'	
BLKSPK		001323'	EXT	BUSYRT		000020		CAPCTY		001660'	
CH1		000071'	EXT	CH2		000073'	EXT	CH3		000075'	EXT
CH4		000077'	EXT	CH5		000101'	EXT	CH6		000103'	EXT
CH7		000105'	EXT	CHCPAR		001000		CHDPAR		002000	
CHKSAV		002114'		CHKSUM		001511'	EXT	CHREC		000010	
CNTDB		000432'	EXT	CONFIG		000170'	EXT	CONFLP		000320'	
CONMES		001005'	EXT	CORLIM		000464'	EXT	CPCT1		001665'	
CPCT2		001674'		CPCT3		001677'		CPCTM1		001707'	
CPOPJ		000741'	EXT	CPOPJ1		001576'	EXT	CRLF		000552'	EXT
CRLFMS		001200'		CTEXT		000410'	EXT	DAT		000005	INT
DATERR		006000		DATLOP		000203'		DCHNPE		002000	
DCPUTR		000717'	EXT	DDE		100000		DDSKPE		004000	
DDTEND		000427'	EXT	DDTSYM		000156'	EXT	DDTX		000161'	EXT
DECIN		000524'	EXT	DECIN1		000260'	EXT	DELET1		000724'	
DELETE		000717'		DESMES		001167'		DEVDAT		000006	INT
DEVERR		211100		DEVIOS		000002	INT	DEVLST		000315'	EXT
DEVNAM		000030	INT	DEVOPR		000416'	EXT	DEVPHY		000414'	EXT
DEVSER		000003	INT	DFGETF		000000	EXT	DFRED		002022'	EXT
DFWNZ1		001241'		DFWRT		002043'	EXT	DFWUNS		001201'	INT
DISKUP		001315'	EXT	DONEFL		000010		DPAR		004000	
DSK		000170		DSKBUF		001246'	EXT	DSKCNT		000000	EXT
DSKDDB		001244'	EXT	DSKFDG		002041'	EXT	DSKINI		000061'	EXT
DTIP		400000		ERROR		000106'	EXT	EXECIS		001150'	
FT2REL	777777	777777	INT	FTCCL	777777	777777		FTCHEC		000000	INT
FTDISK	777777	777777	INT	FTLOGI	777777	777777		FTRC10	777777	777777	INT
FTSWAP	777777	777777	INT	FTTTYS	777777	777777	INT	GET1		000672'	
GET2		000704'		GETCHR		001021'	EXT	GETLIN		000670'	
GETLN1		000731'		ICONM		001002'		ILM		020000	
INLMES		001207'	EXT	INTLOP		000041'		INTLP1		000046'	
INTNUM		000317'	EXT	INTTAB		002325'	EXT	INTTB1		000320'	EXT
IOCONF		001120'		IOS		000000	INT	ITEM		000004	INT
JIFMIN		002132'	EXT	JORN		000135'	EXT	JOBPFI		000001'	EXT
JORSYM		000146'	EXT	JRSTI		000066'		JRSTI1		000034'	
JSAT06		001250'	EXT	K4SWAP		001623'	EXT	LBHIGH		001705'	EXT
LBSAVE		002115'		LENGTH		001153'		LINBUF		001047'	
LINE		000002		LINEP		001046'		LINKDB		000065'	EXT
LINKSR		000000'	INT	LOC40		000067'		LOWSWP		002316'	
MAKEND		000422'	EXT	MAXSWP		001624'	EXT	MDTMDT		002100'	
MFDBLK		001617'	EXT	MOVJMP		000142'		MOVSTP		000141'	INT
MQIN		001256'	EXT	MQOUT		000000	EXT	MSS	010000	000000	
MXK2SW		000527'	EXT	MXKMES		001160'		NODDB		000335'	
NOPICH		000056'		NRDY		220000		NULPDL		000060'	EXT
NUMBIT		001446'	EXT	NUMBLK		001671'	EXT	NUMSAT		001660'	EXT
NXM		010000		NXMEM		000400		OCTIN		000506'	EXT
OCTPNT		000551'	EXT	ONCE		000162'	INT	ONCE2		000424'	
ONCE3		000431'		ONCE4		000342'		ONCE4A		000444'	
ONCE5		000340'		ONCE5A		000465'		ONCE6		000347'	
ONCE7		000355'		ONCE7A		000366'		ONCE8		000417'	
ONCEND		002317'		ONCEOP		000403'		ONCEPD		001026'	
ONCEPN		000020		OPOUT		000734'		OPOUT1		000736'	
OPOUT2		000744'		OPRM		001125'		OTSET		001006'	

SYMBOL TABLE

OUTCHS		000357'	EXT	OVR	000100		PATCH	000163'	EXT
PATSIZ		000163'	INT	PATSYM	000160'	INT	PDP	000003	INT
PLGSW	004000	000000		PRNAME	000330'	EXT	PROG	000007	INT
PSF		010000		PUTCHI	000727'	EXT	QUICK	000466'	
RADX10		000447'	EXT	RCXCCW	002027'	EXT	RCXFIN	002030'	EXT
RCXWNZ		000162'	INT	RDSTA1	001634'		RDSTA2	001640'	
RDSTAT		001625'		RDSTM1	001645'		REF	000467'	
REF1M		000555'		REF2	000500'		REF2M	000567'	
REF3		000514'		REF3A	000532'		REF3M	000574'	
REF4M		000610'		REF5	000545'		REF6	000475'	
REF6M		000625'		REF7M	000630'		REF8M	000636'	
REF9M		000651'		REFER1	002111'		REFERR	002112'	
REFLAG		000540'	EXT	REFRES	001315'		REFSRF	001526'	
REFSRL		001546'		REFST1	001435'		REFST2	001452'	
REFST3		001460'		REFST4	001473'		REFST5	001475'	
REFST6		001506'		REFST7	001510'		REGT30	000151'	
RESETB		000010		RFRSH1	001361'		RFRSHE	001342'	
RFSHME		001347'		RFST3A	001444'		SAT	000000	EXT
SATBK2		001261'	EXT	SATENT	001543'	EXT	SAT05	001306'	EXT
SATPTR		001523'	EXT	SATTOP	001305'	EXT	SATXWD	001620'	EXT
SCNDDB		001006'	EXT	SEGN	000134'	EXT	SEGPTR	000136'	EXT
SENTSZ		001554'	EXT	SETBFI	000670'	EXT	SETFRE	000000	EXT
SETS1		001253'		SETS2	001254'		SETS3	001266'	
SETS3A		001272'		SETS4	001273'		SETS5	001277'	
SETS5B		001301'		SETS5X	001305'		SETS6	001313'	
SETUUO		000156'		SETWNZ	002061'		SFTERR	160600	
SINGLE		000334'		SKIPS	000523'	EXT	SRCHE	200000	
STATBK		000001	INT	STO1	000031'		STPBIT	000020	
STWNZ1		002066'		STWNZ2	002072'		SYSDAT	000174'	EXT
SYSDDT		000153'	EXT	SYSDM	001145'		SYSM	001133'	
SYSPP		000537'	EXT	SYSSIZ	000443'	EXT	T	000001	
T30SYM		000143'	EXT	TAC	000001	INT	TAC1	000002	INT
TCONLN		000371'	EXT	TEM	000010	INT	THSDAT	002104'	EXT
TIBF		001010'	EXT	TIME	002101'	EXT	TIMEM	001113'	
TIMLOP		000251'		TOBF	001012'	EXT	TODATE	001104'	
TSE		040000		TSETBF	001014'	EXT	TSEXEC	001077'	
TSTAPR		000125'		TSTREG	000111'		TTIBUF	001011'	EXT
TTOBUF		001013'	EXT	TWOREG	000303'	EXT	TYI	000771'	
TYO		000751'		TYPL	000000	EXT	USRLIM	000446'	EXT
UUO		000014	INT	UUO2	000070'	EXT	UUO2	000110'	EXT
UUOTRP		000070'	INT	VONCE	000414	INT	WATMSG	002056'	
WLBIT		000000	EXT	WNZBUF	002116'		WNZCLR	002074'	
WNZRD		002013'		WNZRDC	002023'		WNZRDX	002032'	
WNZWAT		002045'		WNZWR	002033'		WNZWTL	002046'	
WRBIT		001000		WRCCWD	000040		WRITLK	001723'	
WRLK2		001737'		WRLK8	001741'		WRLK9	001742'	
WRLKM1		001745'		WRLKM2	001760'		WRPE	000200	
WRSTA1		001566'		WRSTA2	001577'		WRSTA9	001617'	
WRSTAT		001564'		WRSTM1	001603'		WRSTSZ	000006	
WSYNC		001251'	EXT	XJRPFI	000002'	EXT	YESNO	001015'	