RS03/04

DEVICE ROUTINE (MPG)

MD-11-DTRSA-A

EP-DTRSA-A-DL-A

FICHE 1 OF 1

NOV 1976

digital

MADE IN U.S.A.

```
.REM    %

;IDENTIFICATION
;--------------

PRODUCT CODE:   MAINDEC-11-DTRSA-A

PRODUCT NAME:   RH11/RH70 - RS03/RS04 DEVICE
                ROUTINE FOR MPG

DATE:           JULY 1976

MAINTAINED BY:  DIAGNOSTIC ENGINEERING

AUTHOR:         A. W. LEIGH
```



```
        %
```

```
                                .SBTTL  REVISION HISTORY

;
;               JUL 76  DTRSA-A INITIAL RELEASE AS A FULL SUPPORT
;                       DEVICE ROUTINE FOR THE RS03/RS04 DISK.
```

```
                                .SBTTL  STANDARD DEVICE ROUTINE TABLE
                        .TITLE  MAINDEC-11-DTRSA-A  RH11/RH70 - RS03/RS04 DEVICE ROUTINE FOR MPG

                                ;REVISION 'A'

                                ;FILENAME OF "TRSAA0.MPG" ON MPG/XXDP MEDIA

                                ;MACY11:  DTRSA?,DTRSA?/CRF:SYM/DOC=DTRSA?.P11
                                ;LNKX11:  DTRSA?.MPG/B:0=DTRSA?/E
                                ;PAPER TAPE:  PUNCH DTRSA?.MPG/FILE:ELEV


 000000'                        .CSECT  RJP11
                                .DSABL  GBL

                        ;PROGRAM EQUALS
 072460                 ITIME=30000.                    ;TIME COUNT FOR INTERRUPTS
 000100                 NRTRK=64.                       ;NR OF TRACKS
 000100                 NRSEC=64.                       ;NR OF SECTORS
 177776                 PS=177776                       ;PSW
 000016                 REGNUM=14.                      ;NR OF DEVICE REGISTERS
 001011                 STSLUP=1011                     ;LOOP CONTROL FOR "STSTAT" ROUTINE
 000000                 XXXX=0                          ;VALUE TO BE TAYLORED BY DEVICE ROUTINE

                        ;DEVICE REGISTGER NAMES
 000000                 RPCS1=0
 000002                 RPWC=2
 000004                 RPBA=4
 000006                 RPDA=6
 000010                 RPCS2=10
 000012                 RPDS=12
 000014                 RPER=14
 000014                 RPER1=RPER
 000016                 RPAS=16
 000020                 RPLA=20
 000022                 RPDB=22
 000024                 RPMR=24
 000026                 RPDT=26
 000030                 RPBAE=30
 000032                 RPCS3=32


                        ;RSCS1 DEVICE BIT EQUATES
 040000                 TRE=40000
 020000                 MCPE=20000
 004000                 DVA=4000
 002000                 PSEL=2000
 002000                 PORT=PSEL
 000100                 IE=100
 000001                 GO=1

                        ;RSCS2 DEVICE BIT EQUATES
 010000                 NED=10000
 000040                 CLR=40
 000020                 PARITY=20
 000010                 BAI=10
```

```
                                ;RSDS DEVICE BIT EQUATES
100000                          ATA=100000
040000                          ERR=40000
010000                          MOL=10000
000400                          DPR=400

                                ;RSER DEVICE BIT EQUATES
040000                          UNS=40000

                                ;SYSTEM FLAG-WORD BIT DEFINITIONS
000001                          MMVER=1
000002                          USMTPS=2
000010                          CPU70=10

                                ;DISK COMMAND CODES
000010                          DCCODE=10
000030                          SCODE=30
000050                          WCCODE=50
000060                          WCODE=60
000070                          RCODE=70

                                ;DEVICE ROUTINE (DFLGWD) FLAG BITS
100000                          WAITMD=100000           ;WAIT MODE, 0=WAIT
002000                          CORFLG=2000             ;CORRECTION MODE, 0=CORON
000200                          ANYIOI=200              ;ANY I/O HAS BEEN ISSUED
000100                          CMDISU=100              ;I/O COMMAND HAS BEEN ISSUED
000002                          DOTERM=2                ;PROCESS I/O TERMINATION
000001                          IOERR=1                 ;ERROR ON CURRENT I/O
```

```
                                        ;THE FOLLOWING TABLE IS IN THE STANDARDIZED FORMAT REQUIRED
                                        ;TO INTERFACE WITH MPG.

000000' 010132          LOCZ:   .WORD   DVREND-.        ;DEVICE ROUT SIZE IN BYTES
000002' 000000          DFLGWD: .WORD   0               ;DEVICE ROUT FLAGWORD
000004' 000000                  .WORD   0               ;NOT USED
000006' 000000          HEAD:   .WORD   0               ;HEAD # (0 THRU 18.)
        000006'
                        TRAK=HEAD
000010' 000000          SECT:   .WORD   0               ;SECTOR # (0 THRU 21./19.)
000012' 000003          RTRY:   .WORD   3               ;# OF RETRY ATTEMPTS
000014' 000000                  .WORD   0               ;INTERFACE WORD # 5 (NOT USED)
000016' 000000                  .WORD   0               ;INTERFACE WORD # 6 (NOT USED)
000020' 000000          SIZE:   .WORD   0               ;# OF BYTES TRANSFERRED / UNIMAP FLG
000022' 000000          ERRI:   .WORD   0               ;ERROR ON LAST I/O INDICATOR
000024' 172040          DREGAD: .WORD   172040          ;FIRST DEVICE REGISTER ADR
000026' 000204          IVCTAD: .WORD   204             ;INTERRUPT VECTOR ADR
000030' 000240          PSWD:   .WORD   240             ;INT PROC STATUS WORD (BR 5)
000032' 000000                  .WORD   0               ;NOT USED
000034' 001300                  .WORD   HSKEEP-.        ;HOUSEKEEPING ROUT REL ADR
000036' 001350                  .WORD   REPORT-.        ;REPORT ROUT REL ADR
000040' 001764                  .WORD   KILL-.          ;KILL ROUT REL ADR
000042' 001154                  .WORD   DATAER-.        ;DATA ERROR COUNTER REL ADR
000044' 001700                  .WORD   TOUTER-.        ;TIME OUT ERROR ROUT REL ADR
000046' 000000          CIOBSY: .WORD   0               ;I/O BUSY BRANCH ADR
000050' 000000          CUPGER: .WORD   0               ;DEVICE ERROR BRANCH ADR
000052' 000000          ULIST:  .WORD   0               ;USER MODE PRINT ROUTINE BRANCH ADR
000054' 000000          CLIST:  .WORD   0               ;CMND MODE PRINT ROUTINE BRANCH ADR
000056' 000000          BINASC: .WORD   0               ;CONVERT BINARY TO ASCII ROUT BR ADR
000060' 000000          BTASLZ: .WORD   0               ;CONVERT BINARY TO DECIMAL ASCII BR ADR
000062' 000000          DECASC: .WORD   0               ;CONVERT PACKED DECIMAL TO ASCII BR ADR
000064' 000000          CSYSFW: .WORD   0               ;MPG SYSTEM FLAGWORD ADR
000066' 000000          SETVEC: .WORD   0               ;SET INT VECT ROUT BR ADR
000070' 000000          CLRVEC: .WORD   0               ;CLEAR INT VECTOR ROUT BR ADR
000072' 000000          TSTVEC: .WORD   0               ;TEST INT VECTOR ROUT BR ADR
000074' 000000          RTNINT: .WORD   0               ;RETURN FROM INT ROUT BR ADR
000076' 000000          GETBYT: .WORD   0               ;GET DATA BYTE ROUT BR ADR
000100' 000000          PUTBYT: .WORD   0               ;PUT DATA BYTE ROUT BR ADR
000102' 000014                  .WORD   DVREGS-.        ;ADR OF DEVICE REGISTER NAMES
000104' 000136                  .WORD   DVCMDS-.        ;ADR OF DEVICE FUNCTIONS
000106' 000246                  .WORD   DVPKTE-.        ;ADR OF PACK TBL EXTENSION
000110' 000444                  .WORD   DVMVTE-.        ;ADR OF MODEL VECTOR TBL EXTEN.
000112' 000542                  .WORD   DVCPTE-.        ;ADR OF COMPILER TBL EXTEN.
000114' 000716                  .WORD   DVIWST-.        ;ADR OF DEV INTERFACE WD SYM TBL
```

```
                                        .SBTTL  COMPILER TABLES & CONSTANT AREAS


000116' 051522  030503          DVREGS: .ASCII  /RSC1/                  ;VALID DEVICE REGISTER NAMES &
000122' 000000                          .WORD   0                       ;THEIR POSITIONS RELATIVE TO
000124' 051522  041527                  .ASCII  /RSWC/                  ;THE DEVICE REGISTERS BASE ADDRESS.
000130' 000002                          .WORD   2
000132' 051522  040502                  .ASCII  /RSBA/
000136' 000004                          .WORD   4
000140' 051522  040504                  .ASCII  /RSDA/
000144' 000006                          .WORD   6
000146' 051522  031103                  .ASCII  /RSC2/
000152' 000010                          .WORD   10
000154' 051522  051504                  .ASCII  /RSDS/
000160' 000012                          .WORD   12
000162' 051522  051105                  .ASCII  /RSER/
000166' 000014                          .WORD   14
000170' 051522  051501                  .ASCII  /RSAS/
000174' 000016                          .WORD   16
000176' 051522  040514                  .ASCII  /RSLA/
000202' 000020                          .WORD   20
000204' 051522  041104                  .ASCII  /RSDB/
000210' 000022                          .WORD   22
000212' 051522  051115                  .ASCII  /RSMR/
000216' 000024                          .WORD   24
000220' 051522  052104                  .ASCII  /RSDT/
000224' 000026                          .WORD   26
000226' 051522  042501                  .ASCII  /RSAE/
000232' 000030                          .WORD   30
000234' 051522  031503                  .ASCII  /RSC3/
000240' 000032                          .WORD   32
        000242'                 DVREGE= .


000242'     120     211         DVCMDS: .BYTE   120,211               ;VALID DEVICE FUNCTIONS
000244' 002660                          .WORD   READ-.                 ;  FLAG BYTE:
000246'     130     211                 .BYTE   130,211                ;    BIT 7 = NPR DEV
000250' 002704                          .WORD   WRITE-.                ;    BIT 3 = MASSBUS DEV
000252'     376     000                 .BYTE   376,0                  ;    BIT 0 = 2 WORDS FOR ADR
000254' 001736                          .WORD   NOWAIT-.               ;            (18 BIT ADRS)
000256'     375     000                 .BYTE   375,0
000260' 001712                          .WORD   WAIT-.
000262'     374     000                 .BYTE   374,0
000264' 001122                          .WORD   REPORT-.
000266'     373     000                 .BYTE   373,0
000270' 001116                          .WORD   REPORT-.
000272'     370     211                 .BYTE   370,211
000274' 002710                          .WORD   WRCK-.
000276'     365     000                 .BYTE   365,0
000300' 002002                          .WORD   CRESET-.
000302'     364     000                 .BYTE   364,0
000304' 002046                          .WORD   DRESET-.
000306'     363     000                 .BYTE   363,0
000310' 002724                          .WORD   SEARCH-.
000312'     362     000                 .BYTE   362,0
000314' 001542                          .WORD   STEPUP-.
```

```
000316' 000361          000                     .BYTE   361,0
000320' 001622                                  .WORD   STEPDN-.
000322' 000351          000                     .BYTE   351,0
000324' 001676                                  .WORD   APORT-.
000326' 000350          000                     .BYTE   350,0
000330' 001702                                  .WORD   BPORT-.
000332' 000345          000                     .BYTE   345,0
000334' 001706                                  .WORD   ODD-.
000336' 000344          000                     .BYTE   344,0
000340' 001712                                  .WORD   EVEN-.
000342' 000337          000                     .BYTE   337,0
000344' 001716                                  .WORD   BAION-.
000346' 000336          000                     .BYTE   336,0
000350' 001722                                  .WORD   BAIOFF-.
000352' 177777                                  .WORD   177777          ;TABLE TERMINATOR

000354' 047516  040527  052111  DVPKTE: .ASCII  /NOWAIT/               ;PACK TABLE EXTENSION
000362' 000376          000             .BYTE   376,0
000364' 020040  040527  052111          .ASCII  /  WAIT/
000372' 000375          000             .BYTE   375,0
000374' 052123  052101  051525          .ASCII  /STATUS/
000402' 000374          000             .BYTE   374,0
000404' 047503  047525  051524          .ASCII  /COUNTS/
000412' 000373          000             .BYTE   373,0
000414' 020040  042522  045503          .ASCII  /  WRCK/
000422' 000370          000             .BYTE   370,0
000424' 051103  051505  052105          .ASCII  /CRESET/
000432' 000365          000             .BYTE   365,0
000434' 044503  051505  052105          .ASCII  /DRESET/
000442' 000364          000             .BYTE   364,0
000444' 042523  051505  044103          .ASCII  /SEARCH/
000452' 000363          000             .BYTE   363,0
000454' 052123  050124  050125          .ASCII  /STEPUP/
000462' 000362          000             .BYTE   362,0
000464' 052123  050105  047104          .ASCII  /STEPDN/
000472' 000361          000             .BYTE   361,0
000474' 040440  047520  052122          .ASCII  / APORT/
000502' 000351          000             .BYTE   351,0
000504' 041040  047520  052122          .ASCII  / BPORT/
000512' 000350          000             .BYTE   350,0
000514' 020040  047440  042104          .ASCII  /   ODD/
000522' 000345          000             .BYTE   345,0
000524' 020040  053105  047105          .ASCII  /  EVEN/
000532' 000344          000             .BYTE   344,0
000534' 041040  044501  047117          .ASCII  / BAION/
000542' 000337          000             .BYTE   337,0
000544' 040502  047511  043106          .ASCII  /BAIOFF/
000552' 000336          000             .BYTE   336,0

000554' 000376  001064          DVMVTE: .WORD   376,MSFMT1-LOCZ         ;MODEL VECTOR TABLE EXTEN.
000560' 000375  001064                  .WORD   375,MSFMT1-LOCZ
000564' 000374  001064                  .WORD   374,MSFMT1-LOCZ
000570' 000373  001064                  .WORD   373,MSFMT1-LOCZ
000574' 000370  001065                  .WORD   370,MSFMT2-LOCZ
000600' 000365  001064                  .WORD   365,MSFMT1-LOCZ
000604' 000364  001064                  .WORD   364,MSFMT1-LOCZ
```

```
000610' 000363  001064                  .WORD   363,MSFMT1-LOCZ
000614' 000362  001072                  .WORD   362,MSFMT5-LOCZ
000620' 000361  001072                  .WORD   361,MSFMT5-LOCZ
000624' 000351  001064                  .WORD   351,MSFMT1-LOCZ
000630' 000350  001064                  .WORD   350,MSFMT1-LOCZ
000634' 000345  001064                  .WORD   345,MSFMT1-LOCZ
000640' 000344  001064                  .WORD   344,MSFMT1-LOCZ
000644' 000337  001064                  .WORD   337,MSFMT1-LOCZ
000650' 000336  001064                  .WORD   336,MSFMT1-LOCZ


                                ;
                                ;       COMPILER TABLE EXTENSION
                                ;
000654'     003     376         DVCPTE: .BYTE   3,376                   ;NO WAIT
000656' 004537  000012                  .WORD   4537,10.
000662'     003     375                 .BYTE   3,375                   ;WAIT
000664' 004537  000012                  .WORD   4537,10.
000670'     004     374                 .BYTE   4,374                   ;STATUS
000672' 004537  000012  001002          .WORD   4537,10.,1002
000700'     004     373                 .BYTE   4,373                   ;COUNTS
000702' 004537  000012  001001          .WORD   4537,10.,1001
000710'     006     370                 .BYTE   6,370                   ;WRITE CHECK DATA
000712' 004537  000012  000000          .WORD   4537,10.,0,2,2
000720' 000002  000002
000724'     003     365                 .BYTE   3,365                   ;CONTROL RESET
000726' 004537  000012                  .WORD   4537,10.
000732'     003     364                 .BYTE   3,364                   ;DRIVE RESET
000734' 004537  000012                  .WORD   4537,10.
000740'     003     363                 .BYTE   3,363                   ;SEARCH
000742' 004537  000012                  .WORD   4537,10.
000746'     004     362                 .BYTE   4,362                   ;STEP UP
000750' 004537  000012  000000          .WORD   4537,10.,0
000756'     004     361                 .BYTE   4,361                   ;STEP DOWN
000760' 004537  000012  000000          .WORD   4537,10.,0
000766'     003     351                 .BYTE   3,351                   ;A PORT
000770' 004537  000012                  .WORD   4537,10.
000774'     003     350                 .BYTE   3,350                   ;B PORT
000776' 004537  000012                  .WORD   4537,10.
001002'     003     345                 .BYTE   3,345                   ;ODD
001004' 004537  000012                  .WORD   4537,10.
001010'     003     344                 .BYTE   3,344                   ;EVEN
001012' 004537  000012                  .WORD   4537,10.
001016'     003     337                 .BYTE   3,337                   ;BAI ON
001020' 004537  000012                  .WORD   4537,10.
001024'     003     336                 .BYTE   3,336                   ;BAI OFF
001026' 004537  000012                  .WORD   4537,10.


                                ;
                                ;       DEVICE INTERFACE WORD SYMBOL TABLE
                                ;
001032' 051124  045501          DVIWST: .ASCII  /TRAK/
001036' 000006                          .WORD   DEVIW2
001040' 042510  042101                  .ASCII  /HEAD/
001044' 000006                          .WORD   DEVIW2
001046' 042523  052103                  .ASCII  /SECT/
001052' 000010                          .WORD   DEVIW3
```

```
                                        .ASCII  /RTRY/
                                        .WORD   DEVIW4
                                        .WORD   177777          ;END OF TABLE

                                ;
                                ;       MODEL STATEMENT TABLE EXTENSION
                                ;
001064' 000             MSFMT1: .BYTE   0
001065' 377 052101 000377 MSFMT2: .ASCIZ  <377>/AT/<377>
001072' 377     000     MSFMT5: .BYTE   377,0
                                .EVEN


                                ;DEVICE ROUTINE CONSTANTS & EQUATES


        001074'         HSKPST= .
        001074'         ISTAT=  .                       ;STORAGE FOR DEV REG'S AT INT
001074' 000000 000000 000000            .WORD   0,0,0,0,0,0,0,0
001102' 000000 000000 000000
001110' 000000 000000
001114' 000000 000000 000000            .WORD   0,0,0,0,0,0
001122' 000000 000000 000000

001130' 000016          CSTAT:  .BLKW   14.             ;DEV REG CURRENT VALUES STORAGE
001164'                 COUNTS:
001164' 000000          BYRD:   .WORD   0               ;BYTES READ COUNT
001166' 000000                  .WORD   0
001170' 000000          BYWR:   .WORD   0               ;BYTES WRITTEN COUNT
001172' 000000                  .WORD   0
001174' 000000          BYCK:   .WORD   0               ;BYTES CHECKED COUNT
001176' 000000                  .WORD   0
001200' 000000          RDCNT:  .WORD   0               ;READ CMND COUNT
001202' 000000          WRCNT:  .WORD   0               ;WRITE CMND COUNT
001204' 000000          CKCNT:  .WORD   0               ;CHECK CMND COUNT
001206' 000000          SKCNT:  .WORD   0               ;SEARCH CMND COUNT
001210' 000000          CTLCNT: .WORD   0               ;CONTROLLER CLEAR COUNT
001212' 000000          DRVCNT: .WORD   0               ;DRIVE CLEAR CMND COUNT
001214' 000000          ERRCNT: .WORD   0               ;DEVICE ERRORS COUNT
001216' 000000          DATAER: .WORD   0               ;DATA/OPERATOR ERRORS COUNT
001220' 000000          DLTCNT: .WORD   0               ;DATA LATE ERRORS
001222' 000000          DTECNT: .WORD   0               ;DRIVE TIMING ERRORS
001224' 000000          DCKCNT: .WORD   0               ;DATA CHECK ERRORS
001226' 000000          WCECNT: .WORD   0               ;WRITE CHECK ERRORS
001230' 000000          RETRYS: .WORD   0               ;# OF RETRIES ON I/O CMNDS
001232' 000000          INTCNT: .WORD   0               ;INTERRUPTS COUNT
        001234'         CNTEND=.
        000024          CNTNUM=CNTEND-COUNTS/2          ;SIZE OF COUNT TABLE

001234' 000000          ERRADR: .WORD   0               ;CURR ADR IN USER PROG
001236' 000000          CNTADR: .WORD   0               ;ADR OF BYTE COUNT TOTALS
001240' 000000          CURFLG: .WORD   0               ;FLAG WORD OF CURR CMND
001242' 000000          CURCMD: .WORD   0               ;CURR CMND CODE
001244' 000000          CURADR: .WORD   0               ;CURR BUS ADDRESS
001246' 000000                  .WORD   0
001250' 000000          CURCNT: .WORD   0               ;NEG WORD CNT FOR CURR CMND
```

```
001252' 000000          FINCNT: .WORD   0               ;FINAL WORD CNT (RPWC)
001254' 000000          CURRTY: .WORD   0               ;CURR RETRY COUNT
001256' 000000          RTRYIP: .WORD   0               ;RETRY IN PROGRESS FLAG
        001260'         HSKPEN= .

001260' 000000          RPCS1V: .WORD   0               ;BASE VALUE FOR RSCS1 REG

001262' 000000          RPCS2V: .WORD   0               ;BASE VALUE FOR RPCS2 REG





001264'                 PATCH:  .REPT   20..            ;PATCH AREA
                                .WORD   0
                                .ENDR
```

```
                                        .SBTTL  RS03/RS04 SUPPORT ROUTINES ENTERED FROM MPG


                                        ;DEVICE ROUTINE HOUSEKEEPING

                                                ;JSR     R5,HSKEEP      S/R CALL
                                                ;.WORD   0 OR 1         0 = DO HSKP PER OPSW
                                                ;                       1 = UNCOND. DO HSKP
                                                ;R2 = PROG'S OPSW
                                                ;DESTROYS R0,R1

001334' 012767  000003  176450  HSKEEP: MOV     #3,RTRY                 ;INIT # OF RETRY ATTEMPTS
001342' 005067  177712                  CLR     RPCS1V                  ;INITIALIZE RPCS1 VALUE
001346' 005067  177710                  CLR     RPCS2V                  ;INITIALIZE RPCS2 VALUE
001352' 005725                          TST     (R5)+                   ;UNCONDITIONALLY DO HSKP?
001354' 001003                          BNE     10$                     ;N,Y-10$
001356' 032702  000004                  BIT     #HSKPEP,R2              ;OPSW SPECIFY EACH PASS HSKP?
001362' 001010                          BNE     30$                     ;Y,N-30$
001364' 010700                  10$:    MOV     PC,R0                   ;SET UP FIRST WD ADR
001366' 062700  177506                  ADD     #HSKPST-.,R0
001372' 012701  000072                  MOV     #HSKPEN-HSKPST/2,R1     ;SET UP # OF WORDS
001376' 005020                  20$:    CLR     (R0)+                   ;HSKP ALL NECESSARY AREAS
001400' 005301                          DEC     R1
001402' 001375                          BNE     20$
001404' 000205                  30$:    RTS     R5                      ;EXIT IN-LINE


                                        ;RS03/RS04 REPORT ROUTINE

                                                ;JSR     R5,REPORT      S/R CALL
                                                ;.WORD   FLGWD          FLAGWORD
                                                ;                         BIT 15 = CMND MODE CALL
                                                ;                         BIT  9 = PROG STMNT CALL
                                                ;                         BIT  1 = DO STATUS REPORT
                                                ;                         BIT  0 = DO COUNTS REPORT

001406' 004067  004272          REPORT: JSR     R0,SAVREG               ;SAVE REG'S R0 - R5
001412' 004767  004320                  JSR     PC,SUPTAD               ;SET UP PROG TBL ADR IN R3
001416' 011504                          MOV     (R5),R4                 ;GET FLAGWORD
001420' 032704  000002                  BIT     #2,R4                   ;GOING TO DO STATUS DISPLAY?
001424' 001403                          BEQ     5$                      ;Y,N-5$
001426' 004567  004324                  JSR     R5,STSTAT               ;GO STORE STATUS REG'S
001432' 177476                          .WORD   CSTAT-.
001434' 032704  177776          5$:     BIT     #177776,R4              ;DISPLAYING CNTS AT END OF
001440' 001012                          BNE     15$                     ;PROG PASS? (Y,N-15$)
001442' 010700                          MOV     PC,R0                   ;SET UP ADR OF CNTS
001444' 062700  177520                  ADD     #COUNTS-.,R0
001450' 012701  000024                  MOV     #CNTNUM,R1              ;GET # OF CNT WORDS
001454' 005720                  10$:    TST     (R0)+                   ;THIS CNT WORD = 0?
001456' 001003                          BNE     15$                     ;Y,N-15$
001460' 005301                          DEC     R1                      ;DECR WORD CNT
001462' 001374                          BNE     10$                     ;CK'ED ALL WORDS? (Y,N-10$)
001464' 000477                          BR      DVREX                   ;GO TO EXIT -- ALL CNTS ARE 0'S
001466' 004767  004374          15$:    JSR     PC,DEVID                ;DISPLAY DEVICE I.D.
001472' 032704  000002                  BIT     #2,R4                   ;DOING STATUS DISPLAY?
001476' 001432                          BEQ     DISCNT                  ;Y,N-DISCNT
```

```
001500' 010700                   MOV     PC,R0                ;SET UP ADR OF REG'S AT
001502' 062700  177372           ADD     #ISTAT-.,R0          ;LAST INT
001506' 012701  000016           MOV     #REGNUM,R1           ;SET UP # OF REG'S
001512' 005720          20$:     TST     (R0)+                ;ALL REG'S = 0?
001514' 001003                   BNE     30$                  ;N,Y-40$
001516' 005301                   DEC     R1
001520' 001374                   BNE     20$
001522' 000407                   BR      40$
001524' 004567  004710  30$:     JSR     R5,PRINT             ;ISSUE 'AT LAST INT' MSG
001530' 005031                   .WORD   ATIMSG-.
001532' 000014                   .WORD   12.
001534' 004567  004442           JSR     R5,DISPST            ;GO DISPLAY STATUS AT LAST INT
001540' 177334                   .WORD   ISTAT-.
001542' 004567  004672  40$:     JSR     R5,PRINT             ;ISSUE 'CURRENTLY' MSG
001546' 005027                   .WORD   CURMSG-.
001550' 000012                   .WORD   10.
001552' 004567  004424           JSR     R5,DISPST            ;GO DISPLAY CURRENT STATUS
001556' 177352                   .WORD   CSTAT-.
001560' 004767  004616           JSR     PC,PRTIWD            ;GO DISPLAY INFO WORDS
001564' 032704  000001  DISCNT:  BIT     #1,R4                ;DISPLAY COUNTS?
001570' 001431                   BEQ     RPTEND               ;Y,N-RPTEND
001572' 012700  000024           MOV     #CNTNUM,R0           ;SET UP # OF WORDS
001576' 010701                   MOV     PC,R1                ;SET UP ADR OF CNTS
001600' 062701  177364           ADD     #COUNTS-.,R1
001604' 010702                   MOV     PC,R2                ;SET UP TBL ADR
001606' 062702  000066           ADD     #REPTBL-.,R2
001612' 012267  000012  RPTLP:   MOV     (R2)+,RPTBAS         ;MOV MSG ADR TO S/R LINKAGE
001616' 004067  004062           JSR     R0,SAVREG            ;SAVE ALL REG'S
001622' 011100                   MOV     (R1),R0              ;GET CURRENT COUNT
001624' 004577  176226           JSR     R5,@BINASC           ;CONVERT IT TO ASCII
001630' 000000          RPTBAS:  .WORD   XXXX
001632' 004067  004062           JSR     R0,RESREG            ;RESTORE REG'S
001636' 005721                   TST     (R1)+                ;POINT AT NXT CNT
001640' 005300                   DEC     R0                   ;DONE ALL WORDS?
001642' 001363                   BNE     RPTLP                ;Y,N-RPTLP
001644' 004567  004570           JSR     R5,PRINT             ;GO ISSUE COUNTS MSG
001650' 005060                   .WORD   CNTSMG-.
001652' 000445                   .WORD   CNTSEN-CNTSMG
001654' 004567  004560  RPTEND:  JSR     R5,PRINT             ;ISSUE "END OF REPORT" MSG
001660' 004727                   .WORD   RENDMG-.
001662' 177761                   .WORD   -15.
001664' 004067  004030  DVREX:   JSR     R0,RESREG            ;RESTORE REGISTERS
001670' 005725                   TST     (R5)+                ;SET UP RETURN POINT
001672' 000205                   RTS     R5                   ;EXIT IN-LINE


001674' 005114          REPTBL:  .WORD   BCMRD-RPTBAS
001676' 005122                   .WORD   BCMRD+6-RPTBAS
001700' 005137                   .WORD   BCMWR-RPTBAS
001702' 005145                   .WORD   BCMWR+6-RPTBAS
001704' 005163                   .WORD   BCMCK-RPTBAS
001706' 005171                   .WORD   BCMCK+6-RPTBAS
001710' 005216                   .WORD   CMDCRD-RPTBAS
001712' 005232                   .WORD   CMDCWR-RPTBAS
001714' 005247                   .WORD   CMDCCK-RPTBAS
001716' 005265                   .WORD   CMDCSK-RPTBAS
```

```
                                        .WORD   CMDDRV-RPTBAS
                                        .WORD   CMDCMS-RPTBAS
                                        .WORD   CNTCEC-RPTBAS
                                        .WORD   CNTDER-RPTBAS
                                        .WORD   CNTDLT-RPTBAS
                                        .WORD   CNTHCE-RPTBAS
                                        .WORD   CNTDCK-RPTBAS
                                        .WORD   CNTWCE-RPTBAS
                                        .WORD   CNTRTY-RPTBAS
                                        .WORD   CNTINT-RPTBAS



                                ;TIMEOUT ERROR ROUTINE

                                        ;JSR    R5,TOUTER       S/R CALL

TOUTER: JSR     R0,SAVREG               ;SAVE ALL REGISTERS
        JSR     PC,SUPTAD               ;SET UP RPCS1 & PROG TBL ADR'S
        JSR     R5,STSTAT               ;STORE CURRENT STATUS
        .WORD   CSTAT-.
        JSR     R5,TVECT                ;CK IF I HAVE VECTOR CONTROL
        BR      10$                     ;BR IF I DON'T
        BICB    #100,(R4)               ;RESET INT ENABLE
        JSR     PC,RINTV                ;RESET THE INTERRUPT VECTOR
10$:    BIC     #WT4IOT,(R3)            ;RESET WAITING FOR I/O FLG
        JSR     R5,ERRCS1               ;ISSUE I/O TIMEOUT ERROR MSG
        .WORD   IOTO-ERMBAS
        JSR     R0,RESREG               ;RESTORE REGISTERS
        MOV     (SP)+,R5                ;REMOVE RETURN ADR
        JMP     @CUPGER                 ;GO TO ERROR EXIT



                                ;KILL USER PROGRAM ROUTINE

                                        ;JSR    R5,KILL         S/R CALL
                                        ;R3 MUST CONTAIN PROG TBL ADR
                                        ;DESTROYS R0,R1

KILL:   MOV     DREGAD,R1               ;GET DEV REG ADR
        JSR     R5,TVECT                ;DO I HAVE VECTOR CONTROL?
        BR      KILLEX                  ;BR IF I DON'T
        BITB    #100,(R1)               ;IS INT ENABLE SET?
        BEQ     10$                     ;Y,N-10$
        BICB    #100,(R1)               ;RESET INT ENABLE
10$:    JSR     PC,RINTV                ;RESET INT VECTOR INFO
KILLEX: RTS     R5                      ;EXIT IN-LINE
```

```
                                        .SBTTL  RS03/RS04 NON-I/O FUNCTION ROUTINES


                                        ;"STEPUP" FUNCTION ROUTINE

                                                ;JSR    R5,STEPUP       FUNCTION CALL
                                                ;.WORD  NBR             INCREMENT FACTOR

002056' 004767  000036          STEPUP: JSR     PC,STPCOM               ;DO COMMON SETUP
002062' 062502                          ADD     (R5)+,R2                ;ADD INCR VALUE TO SECT #
002064' 020203                  STEPU2: CMP     R2,R3                   ;IS SECT # IN RANGE?
002066' 103403                          BLO     STEPU1                  ;YES
002070' 160302                          SUB     R3,R2                   ;ADJ SECT # DOWNWARDS
002072' 005201                          INC     R1                      ;ADD 1 TO HEAD #
002074' 000773                          BR      STEPU2                  ;CHECK IT AGAIN
002076' 020104                  STEPU1: CMP     R1,R4                   ;IS TRAK # IN RANGE?
002100' 103402                          BLO     STEPEX                  ;N,Y-STEPEX
002102' 160401                          SUB     R4,R1                   ;ADJ TRACK # DOWNWARDS
002104' 000774                          BR      STEPU1                  ;CHECK IT AGAIN
002106'                         STEPEX:
002106' 010167  175674                  MOV     R1,HEAD
002112' 010267  175672                  MOV     R2,SECT
002116' 000205                          RTS     R5                      ;EXIT TO USER PROG


002120'                         STPCOM:
002120' 016701  175662                  MOV     TRAK,R1                 ;GET STARTING TRACK
002124' 016702  175660                  MOV     SECT,R2                 ;  AND SECTOR
002130' 012704  000100                  MOV     #NRTRK,R4               ;GET NR TRACKS
002134' 012703  000100                  MOV     #NRSEC,R3               ;  AND NR SECTORS
002140' 000207                          RTS     PC


                                        ;"STEPDN" FUNCTION ROUTINE

                                                ;JSR    R5,STEPDN       FUNCTION CALL
                                                ;.WORD  NBR             DECREMENT FACTOR

002142' 004767  177752          STEPDN: JSR     PC,STPCOM               ;DO COMMON SETUP
002146' 162502                          SUB     (R5)+,R2                ;SUB DECR FACTOR FROM SECT #
002150' 020203                  STEPD2: CMP     R2,R3                   ;IS SECT # IN RANGE?
002152' 103403                          BLO     STEPD1                  ;YES
002154' 060302                          ADD     R3,R2                   ;ADJ SECT # UPWARDS
002156' 005301                          DEC     R1                      ;DECR HEAD # BY 1
002160' 000773                          BR      STEPD2                  ;CHECK IT AGAIN
002162' 020104                  STEPD1: CMP     R1,R4                   ;IS TRACK # IN RANGE?
002164' 103750                          BLO     STEPEX                  ;N,Y-STEPEX
002166' 060401                          ADD     R4,R1                   ;ADJ IT UPWARDS
002170' 000774                          BR      ST                      ;CHECK IT AGAIN


                                        ;"WAIT" FUNCTION ROUTINE

                                                ;JSR    R5,WAIT         FUNCTION CALL

002172' 042767  100000  175602  WAIT:   BIC     #WAITMD,DFLGWD          ;RESET THE "NOWAIT" FLAG
```

```
                                        JSR     PC,CKDBSY               ;WAIT IF BUSY & DO TERMINATION
                                        JSR     PC,RINTV                ;RESET THE INTERRUPT VECTOR
                                        RTS     R5                      ;EXIT IN-LINE


                                ;"NOWAIT" FUNCTION ROUTINE

                                        ;JSR     R5,NOWAIT              FUNCTION CALL

                                NOWAIT: BIS     #WAITMD,DFLGWD          ;SET THE "NOWAIT" FLAG
                                        RTS     R5                      ;EXIT IN-LINE


                                ;"APORT" FUNCTION ROUTINE

                                        ;JSR     R5,APORT               FUNCTION CALL

                                APORT:  BIC     #PORT,RPCS1V            ;RESET THE PORT BIT
                                        RTS     R5                      ;EXIT IN-LINE


                                ;"BPORT" FUNCTION ROUTINE

                                        ;JSR     R5,BPORT               FUNCTION CALL

                                BPORT:  BIS     #PORT,RPCS1V            ;SET THE PORT BIT TO B PORT
                                        RTS     R5                      ;EXIT IN-LINE


                                ;"ODD" FUNCTION ROUTINE

                                        ;JSR     R5,ODD                 FUNCTION CALL

                                ODD:    BIC     #PARITY,RPCS2V          ;RESET THE PARITY BIT
                                        RTS     R5                      ;EXIT IN-LINE
```

```
                                        ;"EVEN" FUNCTION ROUTINE
                                        ;
                                        ;       JSR     R5,EVEN         FUNCTION CALL
                                        ;
002252' 052767  000020  177002  EVEN:   BIS     #PARITY,RPCS2V          ;SET THE PARITY BIT FOR EVEN
002260' 000205                          RTS     R5                      ;EXIT IN-LINE


                                        ;"BAION" FUNCTION ROUTINE
                                        ;
                                        ;       JSR     R5,BAION        FUNCTION CALL
                                        ;
002262' 052767  000010  176772  BAION:  BIS     #BAI,RPCS2V             ;SET THE BAI BIT
002270' 000205                          RTS     R5                      ;EXIT IN-LINE


                                        ;"BAIOFF" FUNCTION ROUTINE
                                        ;
                                        ;       JSR     R5,BAIOFF       FUNCTION CALL
                                        ;
002272' 042767  000010  176762  BAIOFF: BIC     #BAI,RPCS2V             ;RESET THE BAI BIT
002300' 000205                          RTS     R5                      ;EXIT IN-LINE
```

```
                                        .SBTTL  RS03/RS04 NON-INTERRUPT TYPE I/O FUNCTION ROUTINES


                                        ;"CRESET" FUNCTION ROUTINE

                                                ;JSR    R5,CRESET       FUNCTION CALL

002302' 004767  001722          CRESET: JSR     PC,CKDBSY               ;GO CK IF DEV IS BUSY
002306' 005267  176676                  INC     CTLCNT                  ;COUNT CRESET COMMANDS
002312' 005067  175504                  CLR     ERRI                    ;RESET THE ERROR INDICATOR
002316' 005000                          CLR     R0                      ;INITIALIZE TIME OUT COUNT
002320' 052764  000040  000010          BIS     #CLR,RPCS2(R4)          ;PERFORM CLEAR FUNCTION
002326' 105714                  10$:    TSTB    (R4)                    ;READY SET YET?
002330' 100407                          BMI     20$                     ;N,Y-20$
002332' 005300                          DEC     R0                      ;TIMED OUT?
002334' 100774                          BMI     10$                     ;Y,N-10$
002336' 004567  002220                  JSR     R5,ERRCS                ;GO ISSUE CRESET TIMEOUT ERROR
002342' 002661                          .WORD   CRTO-ERMBAS
002344' 000177  175500                  JMP     @CUPGER                 ;GO TO ERR RET POINT IN MPG
002350' 000205                  20$:    RTS     R5                      ;EXIT IN-LINE TO USER'S PROG


                                        ;"DRESET" FUNCTION ROUTINE

                                                ;R5 = USER PROGRAM RETURN ADR
                                                ;R4 = RPCS1 ADR
                                                ;R3 = PROG TBL ADR
                                                ;R2 = FUNCTION'S COMMAND CODE
                                                ;JSR    R5,DRESET       FUNCTION CALL

002352' 012702  000011          DRESET: MOV     #DCCODE!GO,R2           ;SET UP DRESET CMND CODE
002356' 004767  001646                  JSR     PC,CKDBSY               ;SEE IF DEVICE IS BUSY
002362' 005267  176624                  INC     DRVCNT                  ;COUNT DRESET COMMANDS
002366' 004767  000070                  JSR     PC,ACQHSK               ;ACQUIRE & HOUSEKEEP THE DISK
002372' 052767  000300  175402          BIS     #CMDISU+ANYIOI,DFLGWD   ;SET CMND ISSUED FLAGS
002400' 010214                          MOV     R2,(R4)                 ;ISSUE SPECIFIED CMND
002402' 012700  000024                  MOV     #20.,R0                 ;SET UP DELAY CNT
002406' 005300                  30$:    DEC     R0                      ;DELAY FOR A FEW MICROSECONDS
002410' 001376                          BNE     30$
002412' 005714                          TST     (R4)                    ;IS 'SC' ERROR BIT SET?
002414' 100021                          BPL     50$                     ;Y,N-50$
002416' 032714  060000                  BIT     #TRE+MCPE,(R4)          ;TRE OR MCPE ERR BITS SET?
002422' 001011                          BNE     40$                     ;N,Y-40$
002424' 032764  040000  000012          BIT     #ERR,RPDS(R4)           ;ERROR SUMMARY BIT SET?
002432' 001005                          BNE     40$                     ;N,Y-40$
002434' 016400  000016                  MOV     RPAS(R4),R0             ;GET ATA SUMMARY REG
002440' 036700  000454                  BIT     MYATA,R0                ;IS IT MY ATA LINE?
002444' 001405                          BEQ     50$                     ;Y,N-50$
002446' 004567  002110          40$:    JSR     R5,ERRCS                ;REPORT NON-INT TERM ERROR
002452' 003015                          .WORD   NOITER-ERMBAS
002454' 000177  175370                  JMP     @CUPGER                 ;GO TO MPG'S ERROR RETURN POINT
002460' 000205                  50$:    RTS     R5                      ;EXIT TO USER PROGRAM
```

```
                                        ;HOUSEKEEP THE DISK
                                        ;
                                        ;JSR    PC,ACQHSK       S/R CALL
                                        ;
                                        ;R5 = ADR AFTER USER PROG JSR
                                        ;R4 = RPCS1 ADR
                                        ;R3 = PROG TBL ADR
                                        ;
                                        ;DESTROYS R0
                                        ;
002462' 010146          ACQHSK: MOV     R1,-(SP)                ;SAVE R1
002464'                 ACQRTY:
002464' 012763 072460 000030    MOV     #ITIME,PTOCNT(R3)       ;SETUP INTERRUPT TIMER
002472' 042767 000100 175302    BIC     #CMDISU,DFLGWD          ;HSKP FLAG BITS
002500' 116300 000035           MOVB    PCURDV(R3),R0           ;GET MY UNIT #
002504' 020027 000007           CMP     R0,#7                   ;VALID UNIT #?
002510' 101411                  BLOS    10$                     ;N,Y-10$
002512' 012767 003134 000336    MOV     #INVDVN-ERMBAS,ACQEAD   ;SET UP ADR OF INV UNIT # ERR MSG
002520' 005267 176472           INC     DATAER                  ;COUNT OPERATOR ERROR
002524' 005367 176464           DEC     ERRCNT
002530' 000167 000312           JMP     ACQERR                  ;GO REPORT THE ERROR
002534' 010001          10$:    MOV     R0,R1                   ;GET DISPLACEMENT INTO
002536' 006301                  ASL     R1                      ;THE ATA TABLE FOR
002540' 060701                  ADD     PC,R1                   ;THIS UNIT #
002542' 062701 000336           ADD     #ATATBL-.,R1
002546' 112167 000346           MOVB    (R1)+,MYATA             ;STORE ATA BIT MASKS FOR
002552' 111167 000344           MOVB    (R1),OTHATA             ;THIS UNIT #
002556' 056700 176500           BIS     RPCS2V,R0               ;SET PAT & BAI BITS IN UNIT #
002562' 005001                  CLR     R1                      ;HSKP 1ST TIME FLAG
002564' 010064 000010   12$:     MOV     R0,RPCS2(R4)            ;MOVE UNIT # TO RH11
002570' 005714                  TST     (R4)                    ;SELECT THE DRIVE
002572' 032764 010000 000010    BIT     #NED,RPCS2(R4)          ;NON-EXISTENT DRIVE?
002600' 001413                  BEQ     GOTDSK                  ;NO
002602' 005701                  TST     R1                      ;FIRST TIME?
002604' 001005                  BNE     16$                     ;Y,N-16$
002606' 005201                  INC     R1                      ;RESET FIRST TIME
002610' 112764 000100 000001    MOVB    #100,1(R4)              ;DO RH11 ERROR CLEAR
002616' 000762                  BR      12$                     ;GO LOAD UNIT # AGAIN
002620' 012767 002716 000230 16$:       MOV     #NONEXD-ERMBAS,ACQEAD   ;SET UP ADR OF NON-EXIST DRIVE ERR MSG
002626' 000507                  BR      ACQERR                  ;GO REPORT THE ERROR
002630' 012763 072460 000030 GOTDSK:    MOV     #ITIME,PTOCNT(R3)       ;SETUP INTERRUPT TIMER
002636' 032764 010000 000012    BIT     #MOL,RPDS(R4)           ;DISK ON-LINE?
002644' 001004                  BNE     40$                     ;N,Y-40$
002646' 012767 002774 000202    MOV     #OFFLIN-ERMBAS,ACQEAD   ;SET UP ADR OF DISK OFFLINE ERR MSG
002654' 000474                  BR      ACQERR                  ;GO REPORT THE ERROR
002656' 005001          40$:    CLR     R1                      ;RESET ERROR LOOP CNT
002660' 005714          50$:    TST     (R4)                    ;IS 'SC' ERROR BIT SET?
002662' 100062                  BPL     100$                    ;Y,N-100$
002664' 032764 040000 000014    BIT     #UNS,RPER1(R4)          ;IS THERE AN UNSAFE ERROR?
002672' 001033                  BNE     80$                     ;N,Y-80$
002674' 020127 000005           CMP     R1,#5                   ;5TH TIME THRU ON THIS ERROR?
002700' 001004                  BNE     60$                     ;Y,N-60$
002702' 012767 002750 000146    MOV     #INITDE-ERMBAS,ACQEAD   ;SET UP ADR OF INITIATION ERR MSG
002710' 000456                  BR      ACQERR                  ;GO REPORT THE ERROR
002712' 005201          60$:    INC     R1                      ;ADD 1 TO ERROR LOOP CNT
002714' 016400 000016           MOV     RPAS(R4),R0             ;GET ATA REG
```

```
                                                TSTB    R0              ;IS THERE AN ATTENTION?
                                                BEQ     90$             ;Y,N-90$
                                                BIT     OTHATA,R0       ;MY ATA ONLY?
                                                BEQ     90$             ;N,Y-90$
                                                BIT     MYATA,R0        ;MY ATA LINE ALSO?
                                                BNE     70$             ;N,Y-70$
                                                MOV     R0,RPAS(R4)     ;RESET OTHER ATA LINES
                                                MOVB    #100,1(R4)      ;DO RH11 ERROR CLEAR
                                                BR      50$             ;GO CK 'SC' AGAIN
                                        70$:    MOV     R0,RPAS(R4)     ;RESET OTHER ATA LINES
                                                BR      90$             ;GO CLEAR MY DRIVE
                                        80$:    TST     ERRI            ;ERROR ON PREVIOUS I/O?
                                                BNE     90$             ;N,Y-90$
                                                BIT     #ANYIOI,DFLGWD  ;ANY I/O CMNDS BEEN ISSUED?
                                                BEQ     90$             ;Y,N-90$
                                                MOV     #INITUS-ERMBAS,ACQEAD   ;SET UP ADR OF UNSAFE ERROR MSG
                                                BR      ACQERR          ;GO REPORT THE ERROR
                                        90$:    CLR     ERRI            ;RESET PREV I/O ERR FLAG
                                                BIS     #ANYIOI,DFLGWD  ;SET FLAG TO PREVENT LOOP
                                                MOV     #TRE!DCCODE!GO,(R4)     ;DO RH11 & DRIVE CLEAR
                                                BR      50$             ;GO CK 'SC' AGAIN
                                        100$:   CLR     ERRI            ;HSKP ERROR INDICATOR
                                                BIC     #IOERR,DFLGWD   ; & FLAGS
                                                MOV     (SP)+,R1        ;RESTORE R1
                                                RTS     PC              ;EXIT IN-LINE

                                        ACQERR: MOV     R1,-(SP)        ;SAVE R1 & R2
                                                MOV     R2,-(SP)
                                                JSR     R5,ERRCS        ;STORE CURR STATUS & REPORT
                                        ACQEAD: .WORD   INITDE-ERMBAS   ;THE ERROR
                                        ACQERC: MOV     (SP)+,R2        ;RESTORE R1 & R2
                                                MOV     (SP)+,R1
                                                JSR     R5,@CUPGER      ;GO TO MPG'S ERROR RETURN POINT
                                        ACQGDK: JSR     PC,CKDBSY       ;GO CK IF DEV IS BUSY NOW
                                                JMP     ACQRTY          ;GO TRY AGAIN


                                        ATATBL: .BYTE   001,376
                                                .BYTE   002,375
                                                .BYTE   004,373
                                                .BYTE   010,367
                                                .BYTE   020,357
                                                .BYTE   040,337
                                                .BYTE   100,277
                                                .BYTE   200,177

                                        MYATA:  .WORD   0
                                        OTHATA: .WORD   0
```

```
                                        .SBTTL  RS03/RS04 INTERRUPT TYPE I/O FUNCTION ROUTINES


                                        ;"READ" FUNCTION ROUTINE

                                                ;JSR     R5,READ         FUNCTION CALL
                                                ;.WORD   ADR             DATA ADDRESS (BITS 16 - 21)
                                                ;.WORD   ADR             DATA ADDRESS (BITS 0 - 15)
                                                ;.WORD   CNT             BYTE COUNT
                                                ;.WORD   DEV             (NOT USED)

003124' 012702  000171          READ:   MOV     #RCODE!IE!GO,R2         ;GET READ DATA COMMAND
003130' 012701  000235                  MOV     #235,R1                 ;SETUP CMND FLAG WORD
003134' 004767  001070                  JSR     PC,CKDBSY               ;GO CK IF DEV IS BUSY
003140' 005267  176034                  INC     RDCNT                   ;ADD 1 TO READ CMND CNT
003144' 010700                          MOV     PC,R0                   ;SET UP ADR OF BYTES READ CNT
003146' 062700  176020                  ADD     #BYRD+2-.,R0
003152' 000441                          BR      CMDCOM                  ;GO TO CMND COMMON PROCESSING


                                        ;"WRITE" FUNCTION ROUTINE

                                                ;JSR     R5,WRITE        FUNCTION CALL
                                                ;.WORD   ADR             DATA ADDRESS (BITS 16 - 21)
                                                ;.WORD   ADR             DATA ADDRESS (BITS 0 - 15)
                                                ;.WORD   CNT             BYTE COUNT
                                                ;.WORD   DEV             (NOT USED)

003154' 012702  000161          WRITE:  MOV     #WCODE!IE!GO,R2         ;GET WRITE DATA COMMAND
003160' 012701  000235                  MOV     #235,R1                 ;SET UP CMND FLAG WORD
003164' 004767  001040                  JSR     PC,CKDBSY               ;GO CK IF DEV IS BUSY
003170' 005267  176006                  INC     WRCNT                   ;ADD 1 TO WRITE CMND CNT
003174' 010700                          MOV     PC,R0                   ;SET UP ADR OF BYTES WRITTEN CNT
003176' 062700  175774                  ADD     #BYWR+2-.,R0
003202' 000425                          BR      CMDCOM                  ;GO TO CMND COMMON PROCESSING


                                        ;"WRCK" FUNCTION ROUTINE

                                                ;JSR     R5,WRCK         FUNCTION CALL
                                                ;.WORD   ADR             DATA ADDRESS (BITS 16 - 21)
                                                ;.WORD   ADR             DATA ADDRESS (BITS 0 - 15)
                                                ;.WORD   CNT             BYTE COUNT

003204' 012702  000151          WRCK:   MOV     #WCCODE!IE!GO,R2        ;GET WRITE-CHECK COMMAND
003210' 012701  000236                  MOV     #236,R1                 ;SET UP CMND FLAG WORD
003214' 004767  001010                  JSR     PC,CKDBSY               ;GO CK IF DEV IS BUSY
003220' 005267  175760                  INC     CKCNT                   ;ADD 1 TO CHECK CMND COUNT
003224' 010700                          MOV     PC,R0                   ;SET UP ADR OF BYTES
003226' 062700  175750                  ADD     #BYCK+2-.,R0            ;CHECKED COUNT
003232' 000411                          BR      CMDCOM                  ;GO TO CMND COM PROCESSING
```

```
                                        ;"SEARCH" FUNCTION ROUTINE
                                        ;
                                        ;JSR    R5,SEARCH       ;FUNCTION CALL
                                        ;
003234' 012702  000131          SEARCH: MOV     #SCODE!IE!GO,R2         ;GET SEARCH COMMAND
003240' 012701  000060                  MOV     #060,R1                 ;SET UP CMND FLAG WORD
003244' 004767  000760                  JSR     PC,CKDBSY               ;SEE IF DEVICE IS BUSY
003250' 005267  175732                  INC     SKCNT                   ;COUNT SEARCHES
003254' 000400                          BR      CMDCOM                  ;GO TO CMND COMMON PROCESSING
```

```
                                ;INTERRUPT TYPE I/O FUNCTION COMMON PROCESSING ROUTINE
                                ;
                                ;R4 = ADR OF RPCS1 DEV REG
                                ;R3 = PROG TBL ADR
                                ;R2 = COMMAND CODE
                                ;R1 = COMMAND FLAG WORD
                                ;R0 = ADR OF BYTE COUNT, IF APPLICABLE
                                ;
                                ;CMND FLAGWORD FORMAT:
                                ;
                                ;BIT 7 = 200 = PERFORM RETRIES ON CMND
                                ;BIT 6 = 100 = NOT USED
                                ;BIT 5 = 040 = CMND TERMINATES WITH ATA
                                ;BIT 4 = 020 = SET UP HEAD/SECT #
                                ;BIT 3 = 010 = INCREMENT BYTE COUNTS
                                ;BIT 2 = 004 = DATA TRANSFER CMND
                                ;BIT 1 = 002 = 3 ARGUMENT CMND
                                ;BIT 0 = 001 = 4 ARGUMENT CMND
                                ;
003256' 010067 175754   CMDCOM: MOV     R0,CNTADR       ;SAVE ADR OF BYTE COUNT
003262' 004767 177174           JSR     PC,ACQHSK       ;ACQUIRE & HOUSEKEEP THE DISK
003266' 010167 175746           MOV     R1,CURFLG       ;SAVE FLAGWD FOR TERMINATION
003272' 056702 175762           BIS     RPCS1V,R2       ;SET PORT SELECT BIT IN CMND CODE
003276' 010267 175740           MOV     R2,CURCMD       ;SAVE CURR CMND CODE
003302' 032701 000003           BIT     #3,R1           ;THIS CMND HAVE BUS ADR & WD CNT?
003306' 001416                  BEQ     10$             ;Y,N-10$
003310' 012567 175730           MOV     (R5)+,CURADR    ;STORE 2 WORD BUS ADR
003314' 012567 175726           MOV     (R5)+,CURADR+2
003320' 012500                  MOV     (R5)+,R0        ;GET BYTE COUNT
003322' 000241                  CLC                     ;MAKE IT A WORD COUNT
003324' 006000                  ROR     R0
003326' 005400                  NEG     R0              ;MAKE IT NEGATIVE
003330' 010067 175714           MOV     R0,CURCNT       ;SAVE IT
003334' 032701 000001           BIT     #1,R1           ;THERE A 4TH WORD?
003340' 001401                  BEQ     10$             ;Y,N-10$
003342' 005725                  TST     (R5)+           ;BYPASS IT
003344'                 10$:
003344' 004767 000070           JSR     PC,SUIORG       ;GO SET UP REGS FOR I/O
003350' 016767 174436 175676    MOV     RTRY,CURRTY     ;INITIALIZE RETRY COUNT
003356' 005067 175674           CLR     RTRYIP          ;CLEAR RETRY IN PROGRESS FLAG
003362' 012767 003025 001064    MOV     #IOTERM-ERMBAS,INTEAD ;INIT TERMINATION ERROR MSG
003370' 052767 000002 174404    BIS     #DOTERM,DFLGWD  ;SET THE "PROCESS TERMINATION" FLAG
003376' 052713 000010           BIS     #WT4IOT,(R3)    ;SET WAITING FOR I/O TERM FLAG
003402' 052767 000300 174372    BIS     #CMDISU!ANYIOI,DFLGWD ;SET CMND ISSUED FLAGS
003410' 010214                  MOV     R2,(R4)         ;ISSUE SPECIFIED CMND
003412' 005767 174364           TST     DFLGWD          ;"NOWAIT" BIT SET?
003416' 100003                  BPL     40$             ;Y,N-40$
003420' 042713 000010           BIC     #WT4IOT,(R3)    ;RESET WAITING FOR I/O TERM
003424' 000404                  BR      50$             ;GO TO EXIT
003426' 004577 174414   40$:    JSR     R5,@CIOBSY      ;WAIT FOR I/O TO COMPLETE
003432' 004767 000722           JSR     PC,PROCTM       ;GO PROCESS TERMINATION
003436' 000205          50$:    RTS     R5              ;EXIT IN-LINE TO USER PROG
```

```
                                        ;SET UP DEVICE REGS FOR I/O
                                        ;
                                        ;JSR    PC,SUIORG       S/R CALL
                                        ;
                                        ;R4 = RPCS1 ADR
                                        ;R3 = PROG TBL ADR
                                        ;R2 = CMND CODE
                                        ;R1 = CMND FLAGWORD
                                        ;
                                        ;DESTROYS R0

003440' 032701  000020          SUIORG: BIT     #20,R1                  ;NEED TO SET UP HEAD/SECT?
003444' 001411                          BEQ     10$                     ;Y,N-10$
003446' 016746  174334                  MOV     HEAD,-(SP)              ;GET HEAD #
003452' 000316                          SWAB    (SP)                    ;PUT IN CORRECT BIT POSITION
003454' 006216                          ASR     (SP)
003456' 006216                          ASR     (SP)
003460' 056716  174324                  BIS     SECT,(SP)               ;  AND MERGE WITH SECTOR
003464' 012664  000006                  MOV     (SP)+,RPDA(R4)          ;LOAD HEAD & SECT #'S
003470' 032701  000004          10$:    BIT     #4,R1                   ;DATA XFER CMND?
003474' 001423                          BEQ     30$                     ;Y,N-30$
003476' 016700  175542                  MOV     CURADR,R0               ;GET HIGH BITS OF ADR
003502' 042700  177774                  BIC     #177774,R0              ;RESET BITS ABOVE A17
003506' 000300                          SWAB    R0                      ;ALIGN BITS A16 & A17
003510' 050002                          BIS     R0,R2                   ;SET THEM INTO CMND CODE WORD
003512' 016764  175530  000004          MOV     CURADR+2,RPBA(R4)       ;LOAD BITS 0-15 OF ADR
003520' 032777  000010  174336          BIT     #CPU70,@#SYSFW          ;RUNNING ON AN 11/70?
003526' 001403                          BEQ     20$                     ;Y,N-20$
003530' 016764  175510  000030          MOV     CURADR,RPBAE(R4)        ;MOVE BITS A16-A21 TO ADR EXT
003536' 016764  175506  000002  20$:    MOV     CURCNT,RPWC(R4)         ;LOAD WORD COUNT
003544' 000207          .       30$:    RTS     PC                      ;EXIT IN-LINE
```

```
                                        .SBTTL  RS03/RS04 INTERRUPT SERVICE ROUTINE

003546' 004067 002132          RHPINT:  JSR     R0,SAVREG              ;SAVE ALL REGISTERS
003552' 004567 002200                   JSR     R5,STSTAT              ;GO STORE ALL DEV REG'S
003556' 175316                          .WORD   ISTAT-.
003560' 005267 175446                   INC     INTCNT                 ;ADD 1 TO INTERRUPT CNT
003564' 004767 002146                   JSR     PC,SUPTAD              ;SET UP PROG TBL & RPCS1 ADR'S
003570' 016701 175444                   MOV     CURFLG,R1              ;GET THIS CMND'S FLGWD
003574' 005714                          TST     (R4)                   ;IS 'SC' BIT SET?
003576' 100426                          BMI     CKSC                   ;N,Y-CKSC
003600' 032701 000040                   BIT     #40,R1                 ;CMND SUPPOSED TO SET ATA?
003604' 001406                          BEQ     CLRWTF                 ;Y,N-CLRWTF
003606' 012767 003053 000640           MOV     #NOATA-ERMBAS,INTEAD   ;SET UP NO ATA ERR MSG ADR
003614' 052767 000001 174160   SETERR:  BIS     #IOERR,DFLGWD          ;SET THE TERMINATION I/O ERR FLAG
003622' 042713 000010          CLRWTF:  BIC     #WT4IOT,(R3)           ;RESET WAITING FOR I/O TERM
003626' 032767 000004 175404           BIT     #4,CURFLG              ;THIS A DATA TRANSFER CMND?
003634' 001403                          BEQ     INTEX                  ;Y,N-INTEX
003636' 016467 000002 175406           MOV     RPWC(R4),FINCNT        ;SAVE FINAL WORD COUNT
003644' 004067 002050          INTEX:   JSR     R0,RESREG              ;RESTORE ALL REGISTERS
003650' 000177 174220                   JMP     @RTNINT                ;EXIT FROM INTERRUPT

003654' 032714 020000          CKSC:    BIT     #MCPE,(R4)             ;MCPE ERROR SET?
003660' 001402                          BEQ     28$                    ;N,Y-HARDER
003662' 000167 000310                   JMP     HARDER
003666' 032714 040000          28$:     BIT     #TRE,(R4)              ;TRE ERROR BIT SET?
003672' 001031                          BNE     ERRFND                 ;N,Y-ERRFND
003674' 032764 040000 000012           BIT     #ERR,RPDS(R4)          ;ERROR SUMMARY BIT SET?
003702' 001025                          BNE     ERRFND                 ;N,Y-ERRFND
003704' 016400 000016                   MOV     RPAS(R4),R0            ;GET ATA REG
003710' 036700 177204                   BIT     MYATA,R0               ;MY ATA BIT SET?
003714' 001011                          BNE     40$                    ;N,Y-40$
003716' 010064 000016                   MOV     R0,RPAS(R4)            ;RESET OTHER ATA BITS
003722' 032701 000004                   BIT     #4,R1                  ;DATA XFER CMND?
003726' 001335                          BNE     CLRWTF                 ;N,Y-CLRWTF
003730' 036764 177164 000016   30$:     BIT     MYATA,RPAS(R4)         ;MY ATA SET?
003736' 001774                          BEQ     30$                    ;Y,N-30$
003740' 032701 000040          40$:     BIT     #40,R1                 ;CMND SUPPOSED TO SET ATA?
003744' 001326                          BNE     CLRWTF                 ;N,Y-CLRWTF
003746' 012767 003073 000500           MOV     #UXPATA-ERMBAS,INTEAD  ;SET UP UNEXP ATA ERR MSG ADR
003754' 000717                          BR      SETERR                 ;GO TO ERROR EXIT

003756' 032764 037400 000010   ERRFND:  BIT     #037400,RPCS2(R4)      ;UPE/NED/NEM/PGE/MXF/MDPE
003764' 001104                          BNE     HARDER                 ;IN RPCS2? (N,Y-HARDER)
003766' 032764 067017 000014           BIT     #067017,RPER1(R4)      ;UNS/OPI/WLE/IAE/AOE/PAR/RMR/
003774' 001100                          BNE     HARDER                 ;ILR/ILF IN RPER1? (N,Y-HARDER)
003776' 032764 140000 000010           BIT     #140000,RPCS2(R4)      ;DLT OR WCE IN RPCS2?
004004' 001004                          BNE     CKRTRY                 ;N,Y-CKRTRY
004006' 032764 010000 000014           BIT     #010000,RPER1(R4)      ;DTE IN RPER1?
004014' 001500                          BEQ     CKCORR                 ;Y,N-CKCORR
004016' 005767 175234          CKRTRY:  TST     RTRYIP                 ;ALREADY DONE RETRIES ON THIS CMND?
004022' 001036                          BNE     55$                    ;N,Y-55$
004024' 005767 175224                   TST     CURRTY                 ;ARE RETRIES SPECIFIED?
004030' 001462                          BEQ     HARDER                 ;Y,N-HARDER
004032' 032764 100000 000010           BIT     #100000,RPCS2(R4)      ;DLT ERROR?
004040' 001403                          BEQ     42$                    ;Y,N-42$
```

```
1067  004042' 005267 175152                INC     DLTCNT                  ;ADD 1 TO DLT COUNT
1068  004046' 000424                       BR      55$                     ;GO CK RETRY COUNT
1069  004050' 032764 010000 000014 42$:    BIT     #10000,RPER1(R4)        ;DTE ERROR?
1070  004056' 001403                       BEQ     50$                     ;NO
1071  004060' 005267 175136                INC     DTECNT                  ;ADD 1 TO DTE COUNT
1072  004064' 000415                       BR      55$                     ;GO CK RETRY COUNT
1073  004066' 032764 100000 000014 50$:    BIT     #100000,RPER1(R4)       ;DCK ERROR?
1074  004074' 001403                       BEQ     52$                     ;Y,N-52$
1075  004076' 005267 175122                INC     DCKCNT                  ;ADD 1 TO DCK COUNT
1076  004102' 000406                       BR      55$                     ;GO CK RETRY COUNT
1077  004104' 032764 040000 000010 52$:    BIT     #40000,RPCS2(R4)        ;WCE ERROR?
1078  004112' 001402                       BEQ     55$                     ;Y,N-55$
1079  004114' 005267 175106                INC     WCECNT                  ;ADD 1 TO WCE COUNT
1080  004120' 005367 175130        55$:    DEC     CURRTY                  ;DECREMENT RETRY COUNT
1081  004124' 100004                       BPL     60$                     ;CNT EXHAUSTED? (Y,N-60$)
1082  004126' 012767 003112 000320         MOV     #RTYEXH-ERMBAS,INTEAD   ;SET UP EXHAUSTED RETRIES ERR MSG ADR
1083  004134' 000426                       BR      JSETER                  ;GO TO ERROR EXIT
1084  004136' 005267 175066        60$:    INC     RETRYS                  ;ADD 1 TO RETRY TOTAL CNT
1085  004142' 005267 175110                INC     RTRYIP                  ;SET RETRY IN PROGRESS FLAG
1086  004146' 012714 040011                MOV     #TRE!DCCODE!GO,(R4)     ;ISSUE RH11 & DRIVE CLEAR
1087  004152' 016702 175064                MOV     CURCMD,R2               ;GET CURR CMND IN R2
1088  004156' 004767 177256                JSR     PC,SUIORG               ;SET UP DEV REGS
1089  004162' 012763 072460 000030         MOV     #ITIME,PTOCNT(R3)               ;HSKP T/O COUNT
1090  004170' 010214                       MOV     R2,(R4)                 ;RE-ISSUE THE ORIG CMND
1091  004172' 000167 177446                JMP     INTEX                   ;GO TO INT EXIT
1092
1093  004176' 032764 040000 000014 HARDER: BIT     #UNS,RPER1(R4)          ;IS THE UNSAFE BIT SET?
1094  004204' 001402                       BEQ     JSETER                  ;Y,N-SETERR
1095  004206' 112714 000011                MOVB    #DCCODE!GO,(R4)         ;GO TO ERROR EXIT
1096  004212' 000167 177376        JSETER: JMP     SETERR                  ;GO TO ERROR EXIT
1097  004216' 032764 100000 000014 CKCORR: BIT     #100000,RPER1(R4)       ;IS SC SET?
1098  004224' 001764                       BEQ     HARDER                  ;NO
1099  004226' 000673                       BR      CKRTRY                  ;YES
1100
```

```
                                        .SBTTL  SUBROUTINES FOR RS03/RS04 FUNCTION ROUTINES


                                        ;CHECK IF DEVICE IS BUSY AND WAIT IF IT IS

                                                ;JSR    PC,CKDBSY       S/R CALL
                                                ;
                                                ;DESTROYS R0,R3,R4
                                                ;ON EXIT:  R3 = PROG TBL ADR
                                                ;          R4 = RPCS1 ADR

004230' 004767  001502          CKDBSY: JSR     PC,SUPTAD               ;SET UP PROG TBL & RPCS1 ADR'S
004234' 016400  000010.         10$:    MOV     RPCS2(R4),R0            ;GET CURR UNIT #
004240' 010046                          MOV     R0,-(SP)                ;SAVE IT
004242' 042700  177747                  BIC     #177747,R0              ;RESET UNIT # & OTHER BITS
004246' 156300  000035                  BISB    PCURDV(R3),R0           ;SET IN MY UNIT #
004252' 110064  000010                  MOVB    R0,RPCS2(R4)            ;SELECT MY DRIVE
004256' 011400                          MOV     (R4),R0                 ;GET RPCS1 REG
004260' 112664  000010                  MOVB    (SP)+,RPCS2(R4)         ;RESTORE ORIG UNIT #
004264' 032700  000100                  BIT     #100,R0                 ;INT ENABLE ON?
004270' 001403                          BEQ     20$                     ;Y,N-20$
004272' 004577  173550          15$:    JSR     R5,@CIOBSY              ;RELEASE CONTROL
004276' 000756                          BR      10$                     ;GO CK AGAIN
004300'                         20$:
004300' 032767  000002  173474          BIT     #DOTERM,DFLGWD          ;SHOULD PROCESS PREV TERMINATION?
004306' 001403                          BEQ     30$                     ;Y,N-30$
004310' 004767  000044                  JSR     PC,PROCTM               ;GO PROCESS TERMINATION
004314' 000747                          BR      10$                     ;GO CK INT ENABLE AGAIN
004316' 016767  173504  000012  30$:    MOV     IVCTAD,40$              ;STORE INT VECTOR ADR
004324' 016767  173500  000006          MOV     PSWD,45$                ;STORE PROC STATUS WORD
004332' 004577  173530                  JSR     R5,@SETVEC              ;GO SET UP INTERRUPT VECTOR
004336' 000000                  40$:    .WORD   XXXX                    ;  INT VECTOR ADR
004340' 000000                  45$:    .WORD   XXXX                    ;  PSW
004342' 177204                          .WORD   RHPINT-.                ;  REL INT ROUT ADR
004344' 010567  174664                  MOV     R5,ERRADR               ;SAVE CURR USER STMNT ADR
004350' 162767  000004  174656          SUB     #4,ERRADR
004356' 000207                          RTS     PC                      ;EXIT IN-LINE


                                        ;PROCESS TERMINATION OF PREVIOUS I/O FUNCTION

                                                ;JSR    PC,PROCTM       S/R CALL
                                                ;
                                                ;R3 = PROG TABLE ADR
                                                ;
                                                ;DESTROYS R0

004360' 010146                  PROCTM: MOV     R1,-(SP)                ;SAVE R1 & R2
004362' 010246                          MOV     R2,-(SP)
004364' 042767  000002  173410          BIC     #DOTERM,DFLGWD          ;RESET PROCESS TERMINATION FLAG
004372' 032767  000010  174640          BIT     #10,CURFLG              ;INCR BYTE COUNT?
004400' 001417                          BEQ     20$                     ;Y,N-20$
004402' 016700  174642                  MOV     CURCNT,R0               ;GET INITIAL WORD CNT
004406' 005400                          NEG     R0                      ;MAKE IT POSITIVE AGAIN
004410' 016701  174636                  MOV     FINCNT,R1               ;GET FINAL WORD CNT
004414' 100001                          BPL     10$                     ;IS IT NEGATIVE? (Y,N-10$)
```

```
1158  004416' 005401                      NEG     R1                      ;MAKE IT POSITIVE
1159  004420' 160100              10$:    SUB     R1,R0                   ;SUB REMAINING CNT FROM INITIAL CNT
1160  004422' 006300                      ASL     R0                      ;MAKE IT A BYTE CNT
1161  004424' 010067  173370              MOV     R0,SIZE                 ;STORE # OF BYTES ACTUALLY XFERRED
1162  004430' 016701  174602              MOV     CNTADR,R1               ;GET ADR OF BYTE CNT TOTALS
1163  004434' 060011                      ADD     R0,(R1)                 ;ADD IN THIS CNT
1164  004436' 005541                      ADC     -(R1)                   ;UPDATE MOST SIGNF WORD OF CNT
1165  004440' 032767  000001  173334 20$: BIT     #IOERR,DFLGWD           ;WAS THERE AN ERROR?
1166  004446' 001412                      BEQ     PROCEX                  ;Y,N-PROCEX
1167  004450' 004567  000132              JSR     R5,ERRIS                ;GO ISSUE I/O TERMINATION
1168  004454' 003025              INTEAD: .WORD   IO%ERM-ERMBAS           ;ERROR MSG
1169  004456' 004767  000024              JSR     PC,RINTV                ;RESET THE INT VECTOR
1170  004462' 012602                      MOV     (SP)+,R2                ;RESTORE R1 & R2
1171  004464' 012601                      MOV     (SP)+,R1
1172  004466' 004577  173356              JSR     R5,@CUPGER              ;GO TO MPG ERR RETN POINT
1173  004472' 000207                      RTS     PC                      ;RETURN IN-LINE
1174  004474' 004767  000006      PROCEX: JSR     PC,RINTV                ;GO RESET INT VECTOR
1175  004500' 012602                      MOV     (SP)+,R2                ;RESTORE R1 & R2
1176  004502' 012601                      MOV     (SP)+,R1
1177  004504' 000207                      RTS     PC                      ;EXIT IN-LINE
1178
1179
1180                                      ;RESET INTERRUPT VECTOR S/R
1181
1182                                              ;JSR     PC,RINTV       S/R CALL
1183                                              ;R3 MUST CONTAIN PROG TBL ADR
1184                                              ;DESTROYS R0
1185
1186  004506' 004567  000020      RINTV:  JSR     R5,TVECT                ;GO CK IF I HAVE VECTOR CONTROL
1187  004512' 000406                      BR      RINTEX                  ;BR IF I DON'T
1188  004514' 016767  173306  000004      MOV     IVCTAD,10$              ;GET CURR INT VECT ADR
1189  004522' 004577  173342              JSR     R5,@CLRVEC              ;GO HAVE MPG CLEAR IT
1190  004526' 000000              10$:    .WORD   XXXX
1191  004530' 000207              RINTEX: RTS     PC                      ;EXIT IN-LINE
1192
1193
1194                                      ;TEST INTERRUPT VECTOR S/R
1195
1196                                              ;JSR     R5,TVECT       S/R CALL
1197                                              ;BR      LABEL          EXECUTED IF NOT SAME
1198                                              ;R3 MUST CONTAIN PROG TBL ADR
1199                                              ;DESTROYS R0
1200
1201  004532' 016767  173270  000010 TVECT:  MOV     IVCTAD,20$              ;GET CURR INT VECT ADR
1202  004540' 016346  000004              MOV     PFWADR(R3),-(SP)        ;STORE FLGWD ADR TO IDENTIFY ME
1203  004544' 004577  173322              JSR     R5,@TSTVEC              ;DO I HAVE VECTOR CONTROL?
1204  004550' 000000              20$:    .WORD   XXXX                    ;  MPG WILL TELL ME SINCE I CAN'T
1205  004552' 176774                      .WORD   RHPINT-.                ;  GET AT LOWER MEM IF MEM MGMNT
1206  004554' 000401                      BR      TVECTX                  ;BR IF I DONT'T HAVE CNTRL
1207  004556' 005725                      TST     (R5)+                   ;BYPASS BR INST IN S/R CALL
1208  004560' 000205              TVECTX: RTS     R5                      ;EXIT IN-LINE
```

```
                                        ;ERROR INFORMATION DISPLAY S/R
                                        ;
                                        ;JSR     R5,ERRCS        S/R CALL FOR CURR STATUS
                                        ;JSR     R5,ERRCS1       S/R CALL FOR CURR STATUS W/O STORING
                                        ;JSR     R5,ERRIS        S/R CALL FOR INT STATUS
                                        ;.WORD   MSGADR-ERMBAS   REL ADR OF ERROR MSG
                                        ;
                                        ;R3 = PROG TABLE ADR
                                        ;DESTROYS R0,R1,R2
004562' 004567  001170          ERRCS:  JSR     R5,STSTAT               ;STORE CURR STATUS
004566' 174342                  .WORD   CSTAT-.
004570' 012767  173706  000424  ERRCS1: MOV     #CSTAT-ERSTAD,ERSTAD    ;STORE ADR OF CURR STATUS
004576' 012767  174114  000212          MOV     #CSTAT-EBSBAS,EBSTAT
004604' 000406                  BR      ERRCOM                  ;GO TO COMMON POINT
004606' 012767  173652  000406  ERRIS:  MOV     #ISTAT-ERSTAD,ERSTAD    ;STORE ADR OF LAST INT STATUS
004614' 012767  174060  000174          MOV     #ISTAT-EBSBAS,EBSTAT
004622' 012567  000134  ERRCOM: MOV     (R5)+,ERMBAS            ;STORE MSG ADR
004626' 005267  174362          INC     ERRCNT                  ;ADD 1 TO ERROR CNT
004632' 012767  000001  173162  MOV     #1,ERRI                 ;SET THE ERROR INDICATOR
004640' 032763  000400  000002  BIT     #DOERCK,POPSW(R3)       ;SUPPOSED TO DO ERROR CHECKING?
004646' 001004                  BNE     2$                      ;Y,N-2$
004650' 032763  020000  000002  BIT     #PRONER,POPSW(R3)       ;ERROR PRINTING INHIBITED?
004656' 001402                  BEQ     4$                      ;N,Y-4$
004660' 000167  000424  2$:     JMP     ERREX                   ;GO TO EXIT
004664' 010446          4$:     MOV     R4,-(SP)                ;SAVE R4 & R5
004666' 010546                  MOV     R5,-(SP)
004670' 005004                  CLR     R4                      ;SET USER MODE PRINT FLAG
004672' 004767  001170          JSR     PC,DEVID                ;DISPLAY DEVICE I.D.
004676' 032767  000100  173076  BIT     #CMDISU,DFLGWD          ;HAS THE CMND BEEN ISSUED?
004704' 001005                  BNE     6$                      ;N,Y-6$
004706' 004567  001526          JSR     R5,PRINT                ;PRINT THE "BEFORE ISSUING I/O" MSG
004712' 002652                  .WORD   BEFIO-.
004714' 000030                  .WORD   24.
004716' 000404                  BR      8$                      ;GO CALC MSG LNGTH
004720' 004567  001514  6$:     JSR     R5,PRINT                ;PRINT THE "AFTER ISSUING I/O" MSG
004724' 002670                  .WORD   AFTIO-.
004726' 000027                  .WORD   23.
004730' 010700          8$:     MOV     PC,R0                   ;GET START ADR OF ERROR MSG
004732' 062700  000030          ADD     #ERMBAS-.,R0
004736' 061000                  ADD     (R0),R0
004740' 012701  177777          MOV     #-1,R1                  ;INITIALIZE MSG LENGTH
004744' 005201          10$:    INC     R1                      ;ADD 1 TO MSG LENGTH
004746' 105720                  TSTB    (R0)+                   ;MSG TERMINATOR?
004750' 001375                  BNE     10$                     ;Y,N-10$
004752' 010167  000006          MOV     R1,ERMBAS+2             ;STORE MSG LENGTH
004756' 004567  001456          JSR     R5,PRINT                ;PRINT ERROR MSG SPECIFIED
004762' 000000          ERMBAS: .WORD   XXXX
004764' 000000                  .WORD   XXXX
004766' 026727  177770  003134  CMP     ERMBAS,#INVDVN-ERMBAS   ;INVALID UNIT # MSG OR HIGHER?
004774' 103115                  BHIS    ERRSNM                  ;N,Y-ERRSNM
004776' 010701                  MOV     PC,R1                   ;GET ADR OF CODE  AREA IN ERR MSG
005000' 062701  002464          ADD     #CODFLD-.,R1
005004' 010700                  MOV     PC,R0                   ;SET UP ADR OF ERROR CODE TBL
005006' 062700  000304          ADD     #ERCDTB-.,R0
005012' 010702                  MOV     PC,R2                   ;SET UP ADR OF STORED DEV REG'S
```

```
1266  005014' 062702                EBSBAS: ADD     (PC)+,R2
1267  005016' 174114                EBSTAT: .WORD   CSTAT-EBSBAS
1268  005020' 012767 000015 000166         MOV     #13.,70$        ;INITIALIZE MSG LENGTH
1269  005026' 012746 000100                MOV     #64.,-(SP)      ;INITIALIZE CODE FIELD CNT
1270  005032' 012205                15$:    MOV     (R2)+,R5        ;GET NEXT DEV REG WORD
1271  005034' 000305                17$:    SWAB    R5              ;GET DESIRED BYTE IN LOW BYTE
1272  005036' 112004                20$:    MOVB    (R0)+,R4        ;GET FLAG & LENGTH BYTE
1273  005040' 005704                        TST     R4              ;END OF THE CODE TBL?
1274  005042' 001455                        BEQ     60$             ;N,Y-60$
1275  005044' 122704 000377                 CMPB    #377,R4         ;GO TO NXT DEV REG WORD?
1276  005050' 001770                        BEQ     15$             ;N,Y-15$
1277  005052' 122704 000376                 CMPB    #376,R4         ;GO TO NXT BYTE IN DEV REG WORD?
1278  005056' 001766                        BEQ     17$             ;N,Y-17$
1279  005060' 032704 000040                 BIT     #40,R4          ;THIS AN 11/70 ONLY ERROR BIT?
1280  005064' 001405                        BEQ     26$             ;NO
1281  005066' 032777 000010 172770         BIT     #CPU70,@CSYSFW  ;RUNNING ON AN 11/70?
1282  005074' 001411                        BEQ     35$             ;Y,N-35$
1283  005076' 000400                        BR      26$             ;GO CK ERROR BIT
1284  005100' 032704 000100         26$:    BIT     #100,R4         ;BIT VALUE OF 0 = AN ERROR CONDITION?
1285  005104' 001403                        BEQ     30$             ;Y,N-30$
1286  005106' 131005                        BITB    (R0),R5         ;THIS BIT RESET IN DEV REG BYTE?
1287  005110' 001407                        BEQ     40$             ;N,Y-40$
1288  005112' 000402                        BR      35$             ;GO TO NXT TBL ENTRY
1289  005114' 131005                30$:    BITB    (R0),R5         ;THIS ERROR BIT SET IN DEV REG BYTE?
1290  005116' 001004                        BNE     40$             ;N,Y-40$
1291  005120' 042704 177770         35$:    BIC     #177770,R4      ;ISOLATE ENTRY LENGTH
1292  005124' 060400                        ADD     R4,R0           ;POINT AT NXT CODE TBL ENTRY
1293  005126' 000743                        BR      20$             ;GO CK FOR NXT CODE
1294  005130' 042704 177770         40$:    BIC     #177770,R4      ;ISOLATE I.D. NAME LENGTH + 1
1295  005134' 020416                        CMP     R4,(SP)         ;ENOUGH ROOM FOR NAME?
1296  005136' 101017                        BHI     60$             ;Y,N-60$
1297  005140' 060467 000050                 ADD     R4,70$          ;ADJ MSG LENGTH FOR NAME
1298  005144' 005304                        DEC     R4              ;ADJ FOR BIT MASK CHAR
1299  005146' 005200                        INC     R0              ;POINT PAST BIT MASK
1300  005150' 021627 000100                 CMP     (SP),#64.       ;FIRST ERROR CODE IN MSG?
1301  005154' 001403                        BEQ     50$             ;N,Y-50$
1302  005156' 112721 000054                 MOVB    #',,(R1)+       ;MOVE COMMA TO MSG
1303  005162' 005316                        DEC     (SP)            ;ADJ REMAINING ROOM IN MSG
1304  005164' 112021                50$:    MOVB    (R0)+,(R1)+     ;MOVE ERROR CODE TO MSG
1305  005166' 005316                        DEC     (SP)            ;ADJ REMAINING ROOM IN MSG
1306  005170' 005304                        DEC     R4              ;MOVED ALL NAME CHARS?
1307  005172' 001374                        BNE     50$             ;Y,N-50$
1308  005174' 000720                        BR      20$             ;GO CK FOR MORE ERROR BITS
1309  005176' 005004                60$:    CLR     R4              ;SET USER MODE PRINT
1310  005200' 022627 000100                 CMP     (SP)+,#64.      ;ANY ERROR CODES PUT IN MSG?
1311  005204' 001404                        BEQ     80$             ;Y,N-80$
1312  005206' 004567 001226                 JSR     R5,PRINT        ;GO ISSUE ERROR BITS MSG
1313  005212' 002234                        .WORD   DKEMSG-.
1314  005214' 000116                70$:    .WORD   78.
1315  005216' 004567 000760         80$:    JSR     R5,DISPST       ;DISPLAY DEVICE REG'S
1316  005222' 000000                ERSTAD: .WORD   XXXX
1317  005224' 004767 001152                 JSR     PC,PRTIWD       ;DISPLAY TRACK, SECT VALUES
1318  005230' 016300 000022         ERRSNM: MOV     PSRCST(R3),R0   ;GET ADR OF SRC STMNTS
1319  005234' 111001                110$:   MOVB    (R0),R1         ;SAVE STMNT LENGTH
1320  005236' 026067 000004 173770         CMP     4(R0),ERRADR    ;ERROR OCCUR ON THIS STMNT?
1321  005244' 001402                        BEQ     120$            ;N,Y-120$
```

```
                                                ADD     R1,R0                   ;POINT AT NXT STMNT
                                                BR      110$                    ;GO CK NXT STMNT
                                        120$:   TST     (R0)+                   ;SET UP ADR OF STMNT # DATA
                                                MOV     PC,R1                   ;SET UP DATA OUTPUT ADR
                                                ADD     #STMNUM-.,R1
                                                JSR     R5,@DECASC              ;CONVERT IT TO ASCII
                                                MOV     #20040,STMNUM+4         ;SET 2 LOW DIGITS TO SPACES
                                                JSR     R5,PRINT                ;ISSUE STMNT # MSG
                                                .WORD   STMNMG-.
                                                .WORD   -14.
                                                MOV     (SP)+,R5                ;RESTORE R5 & R4
                                                MOV     (SP)+,R4
                                        ERREX:  RTS     R5                      ;EXIT IN-LINE

                                                ;ERROR MESSAGE CODE TABLE

                                                        ;377 = GO TO NEXT DEVICE REGISTER WORD
                                                        ;376 = GO TO NEXT DEVICE REGISTER BYTE
                                                        ;
                                                        ;BYTE 0 CONTAINS FLAG BITS & I.D. NAME LENGTH
                                                        ;
                                                        ;       BITS 0-2 = LENGTH OF BIT MASK + I.D. NAME
                                                        ;       BIT 3 = NOT USED
                                                        ;       BIT 4 = NOT USED
                                                        ;       BIT 5 = 11/70 ONLY ERROR BIT
                                                        ;       BIT 6 = BIT = 0 IS AN ERROR CONDITION
                                                        ;
                                                        ;BYTE 1 IS THE BIT MASK
                                                        ;
                                                        ;BYTES 2 THRU ? ARE THE BIT'S ASCII I.D.

                                        ERCDTB: .ASCII  <003><200>/SC/          ;RPCS1 - BYTE 1
 005316' 040004  051124      105                .ASCII  <004><100>/TRE/
 005323'    005  046440  050103                 .ASCII  <005><040>/MCPE/
 005330'    105
 005331'    104  042010  040526                 .ASCII  <104><010>/DVA/
 005336'    376                                 .BYTE   376
 005337'    104  051200  054504                 .ASCII  <104><200>/RDY/
 005344'    377                                 .BYTE   377
 005345'    377                                 .BYTE   377
 005346'    377                                 .BYTE   377
 005347'    377                                 .BYTE   377
 005350' 100004  046104      124                .ASCII  <004><200>/DLT/         ;RPCS2 - BYTE 1
 005355'    004  053500  042503                 .ASCII  <004><100>/WCE/
 005362' 020004  050125      105                .ASCII  <004><040>/UPE/
 005367'    004  047020  042105                 .ASCII  <004><020>/NED/
 005374' 004004  042516      115                .ASCII  <004><010>/NEM/
 005401'    004  050004  042507                 .ASCII  <004><004>/PGE/
 005406' 001004  054115      106                .ASCII  <004><002>/MXF/
 005413'    005  046401  050104                 .ASCII  <005><001>/MDPE/
 005420'    105
 005421'    377                                 .BYTE   377
 005422' 100004  052101      101                .ASCII  <004><200>/ATA/         ;RPDS - BYTE 1
 005427'    004  042500  051122                 .ASCII  <004><100>/ERR/
 005434' 020004  044520      120                .ASCII  <004><040>/PIP/
```

```
1376  005441'    104  046420  046117          .ASCII  <104><020>/MOL/
1377  005446' 000504  050104     122          .ASCII  <104><001>/DPR/
1378  005453'    376                          .BYTE   376
1379  005454' 100104  051104     131          .ASCII  <104><200>/DRY/          ;RPDS - BYTE 0
1380  005461'    377                          .BYTE   377
1381  005462' 100004  041504     113          .ASCII  <004><200>/DCK/          ;RPER1 - BYTE 1
1382  005467'    004  052500  051516          .ASCII  <004><100>/UNS/
1383  005474' 020004  050117     111          .ASCII  <004><040>/OPI/
1384  005501'    004  042020  042524          .ASCII  <004><020>/DTE/
1385  005506' 004004  046127     105          .ASCII  <004><010>/WLE/
1386  005513'    004  044404  042501          .ASCII  <004><004>/IAE/
1387  005520' 001004  047501     105          .ASCII  <004><002>/AOE/
1388  005525'    376                          .BYTE   376
1389  005526' 004004  040520     122          .ASCII  <004><010>/PAR/
1390  005533'    004  051004  051115          .ASCII  <004><004>/RMR/
1391  005540' 001004  046111     122          .ASCII  <004><002>/ILR/
1392  005545'    004  044401  043114          .ASCII  <004><001>/ILF/
1393  005552'    377                          .BYTE   377
1394  005553'    376                          .BYTE   376
1395  005554' 100005  052101  033501          .ASCII  <005><200>/ATA7/         ;RPAS - BYTE 0
1396  005562' 040005  052101  033101          .ASCII  <005><100>/ATA6/
1397  005570' 020005  052101  032501          .ASCII  <005><040>/ATA5/
1398  005576' 010005  052101  032101          .ASCII  <005><020>/ATA4/
1399  005604' 004005  052101  031501          .ASCII  <005><010>/ATA3/
1400  005612' 002005  052101  031101          .ASCII  <005><004>/ATA2/
1401  005620' 001005  052101  030501          .ASCII  <005><002>/ATA1/
1402  005626' 000405  052101  030101          .ASCII  <005><001>/ATA0/
1403  005634'    377                          .BYTE   377
1404  005635'    377                          .BYTE   377
1405  005636'    377                          .BYTE   377
1406  005637'    377                          .BYTE   377
1407  005640'    377                          .BYTE   377
1408  005641'    377                          .BYTE   377
1409  005642' 100044  050101     105          .ASCII  <044><200>/APE/          ;RPCS3 - BYTE 1 (11/70 ONLY)
1410  005647'    046  042100  042520          .ASCII  <046><100>/DPEOW/
      005654' 053517
1411  005656' 020046  050104  042505          .ASCII  <046><040>/DPEEW/
      005664'    127
1412  005665'    046  053420  042503          .ASCII  <046><020>/WCEOW/
      005672' 053517
1413  005674' 004046  041527  042505          .ASCII  <046><010>/WCEEW/
      005702'    127
1414  005703'    000                          .BYTE   0                        ;TABLE TERMINATOR
1415                                           .EVEN
```

```
                                        .SBTTL  SUBROUTINES FOR RS03/RS04 DEVICE ROUTINE


                                        ;SAVE REGISTERS R0 THRU R5

                                                ;JSR    R0,SAVREG       S/R CALL

005704' 010146                  SAVREG: MOV     R1,-(SP)                ;SAVE R0 THRU R5
005706' 010246                          MOV     R2,-(SP)
005710' 010346                          MOV     R3,-(SP)
005712' 010446                          MOV     R4,-(SP)
005714' 010546                          MOV     R5,-(SP)
005716' 010007                          MOV     R0,PC                   ;EXIT IN-LINE


                                        ;RESTORE REGISTERS R0 THRU R5

                                                ;JSR    R0,RESREG       S/R CALL

005720' 005726                  RESREG: TST     (SP)+                   ;RESTORE R5 THRU R0
005722' 012605                          MOV     (SP)+,R5
005724' 012604                          MOV     (SP)+,R4
005726' 012603                          MOV     (SP)+,R3
005730' 012602                          MOV     (SP)+,R2
005732' 012601                          MOV     (SP)+,R1
005734' 000200                          RTS     R0                      ;EXIT IN-LINE


                                        ;SET PROGRAM'S PROG TABLE ADR IN R3 & RPCS1 ADR IN R4

                                                ;JSR    PC,SUPTAD       S/R CALL

005736' 010703                  SUPTAD: MOV     PC,R3                   ;SET UP LOCATION ZERO ADR
005740' 062703  172040                  ADD     #LOCZ-.,R3
005744' 166303  177776                  SUB     -2(R3),R3               ;SUBTRACT PROG TBL LENGTH
005750' 016704  172050                  MOV     DREGAD,R4               ;GET DEV REG BASE ADR
005754' 000207                          RTS     PC                      ;EXIT IN-LINE


                                        ;STORE DEVICE'S STATUS REGISTERS

                                                ;JSR    R5,STSTAT       S/R CALL
                                                ;.WORD  STADR-.         REL STORAGE ADR
                                                ;DESTROYS R0,R1,R2

005756' 010501                  STSTAT: MOV     R5,R1                   ;GET REL STORAGE ADR & MAKE
005760' 062501                          ADD     (R5)+,R1                ;IT ABSOLUTE
005762' 016700  172036                  MOV     DREGAD,R0               ;GET DEV REG ADR
005766' 010046                          MOV     R0,-(SP)                ;SET UP ADR OF RPCS2 REG
005770' 062716  000010                  ADD     #RPCS2,(SP)             ;FOR LATER USE
005774' 012702  001011                  MOV     #STSLUP,R2              ;SETUP TWO LOOP COUNTS
006000' 032777  000010  172056          BIT     #CPU70,@CSYSFW          ;RUNNING ON AN 11/70?
006006' 001402                          BEQ     10$                     ;Y,N-10$
006010' 062702  001000                  ADD     #1000,R2                ;ALLOW FOR 2 MORE REGS
006014' 012021                  10$:    MOV     (R0)+,(R1)+             ;STORE DEV REG
```

```
1473  006016' 105302                      DECB    R2              ;FINISHED WITH THIS GROUP OF REGS?
1474  006020' 001375                      BNE     10$             ;Y,N-10$
1475  006022' 000302                      SWAB    R2              ;SET UP NEXT LOOP CNT
1476  006024' 001417                      BEQ     30$             ;DONE 2 PASSES? (N,Y-30$)
1477  006026' 010746                      MOV     PC,-(SP)        ;SET UP CURRENT STATUS
1478  006030' 062716  173100              ADD     #CSTAT-.,(SP)   ;STORAGE ADR
1479  006034' 020126                      CMP     R1,(SP)+        ;STORING STATUS FOR INTERRUPT?
1480  006036' 101005                      BHI     15$             ;Y,N-15$
1481  006040' 032736  000200              BIT     #200,@(SP)+     ;OUTPUT READY SET IN RPCS2?
1482  006044' 001403                      BEQ     20$             ;Y,N-20$
1483  006046' 012021                      MOV     (R0)+,(R1)+     ;STORE RSDB CONTENTS
1484  006050' 000761                      BR      10$             ;GO DO SECOND PASS
1485  006052' 005726              15$:    TST     (SP)+           ;TAKE UNUSED ADR OFF STACK
1486  006054' 062700  000002      20$:    ADD     #2,R0           ;BYPASS READ OF RPDB
1487  006060' 005021                      CLR     (R1)+           ;SET ITS STORAGE TO 0'S
1488  006062' 000754                      BR      10$             ;GO DO SECOND PASS
1489  006064' 000205              30$:    RTS     R5              ;EXIT IN-LINE
1490
1491
1492                                      ;DISPLAY DEVICE I.D. & UNIT #
1493
1494                                              ;JSR    PC,DEVID        S/R CALL
1495
1496                                              ;R3 MUST CONTAIN PROG TBL ADR
1497                                              ;DESTROYS R0,R1,R2
1498
1499  006066' 012700  031460      DEVID:  MOV     #"03,R0         ;INITIALIZE TO RS03
1500  006072' 032763  000020  000032      BIT     #20,PMDLCD(R3)  ;IS IT AN RS04?
1501  006100' 001402                      BEQ     10$             ;NO
1502  006102' 012700  032060              MOV     #"04,R0         ;YES
1503  006106' 010067  000522      10$:    MOV     R0,UNITMG+6     ;TAILOR DEV ID MESSAGE
1504  006112' 012767  000026  000056      MOV     #22.,DEVIML     ;INITIALIZE TO NORMAL MS
1505  006120' 116300  000035              MOVB    PCURDV(R3),R0   ;GET CURR UNIT #
1506  006124' 020027  000007              CMP     R0,#7           ;VALID UNIT #?
1507  006130' 101007                      BHI     DEVIIV          ;Y,N-DEVIIV
1508  006132' 004577  171722              JSR     R5,@BTASLZ      ;CONVERT # TO DECIMAL ASCII
1509  006136' 000514                      .WORD   UNASCI-.
1510  006140' 016767  000512  000504      MOV     UNASCI+4,UNASCI ;MOVE ASCII # TO 1ST TWO DIGITS
1511  006146' 000410                      BR      DEVIPR          ;GO ISSUE MSG
1512  006150' 012767  000032  000020  DEVIIV: MOV  #26.,DEVIML    ;SETUP ERROR COND MSG LNGTH
1513  006156' 042700  177400              BIC     #177400,R0      ;RESET HIGH BYTE
1514  006162' 004577  171670              JSR     R5,@BINASC      ;CONVERT BINARY # TO ASCII
1515  006166' 000464                      .WORD   UNASCI-.
1516  006170' 004567  000244      DEVIPR: JSR     R5,PRINT        ;GO ISSUE UNIT # MSG
1517  006174' 000432                      .WORD   UNITMG-.
1518  006176' 000026              DEVIML: .WORD   22.
1519  006200' 000207                      RTS     PC              ;EXIT IN-LINE
```

```
                                        ;TAILOR STATUS MSG & PRINT IT
                                        ;
                                        ;JSR    R5,DISPST       S/R CALL
                                        ;WORD   STATADR-.       REL ADR OF STATUS DATA
                                        ;DESTROYS R0,R1,R2
                                        ;
006202' 010346          DISPST: MOV     R3,-(SP)                ;SAVE R3
006204' 010503                  MOV     R5,R3                   ;GET REL DATA ADR
006206' 062503                  ADD     (R5)+,R3                ;MAKE IT ABS
006210' 010546                  MOV     R5,-(SP)                ;SAVE R5
006212' 010705                  MOV     PC,R5                   ;SET UP ADR OF REG NAMES IN ASCII
006214' 062705  171702          ADD     #DVREGS-.,R5
006220' 012746  000014          MOV     #REGNUM-2,-(SP)         ;SETUP # OF REG TO DISPL
006224' 032777  000010  171632  BIT     #CPU70,@#SYSFW          ;RUNNING ON AN 11/70?
006232' 001402                  BEQ     10$                     ;Y,N-10$
006234' 062716  000002          ADD     #2,(SP)                 ;MAKE IT 14 REGISTERS
006240' 012700  000003  10$:    MOV     #3,R0                   ;SET UP 3 REG LOOP CNT
006244' 010701                  MOV     PC,R1                   ;POINT AT REG NAME IN MSG
006246' 062701  000412          ADD     #DVRGMG-.,R1
006252' 012521          15$:    MOV     (R5)+,(R1)+             ;MOVE REG NAME TO MSG
006254' 012521                  MOV     (R5)+,(R1)+
006256' 005725                  TST     (R5)+                   ;POINT TO NEXT NAME
006260' 062701  000012          ADD     #10.,R1                 ;POINT TO NEXT FIELD IN MSG
006264' 005300                  DEC     R0                      ;DONE 3 REGS?
006266' 001371                  BNE     15$                     ;Y,N-15$
006270' 012300                  MOV     (R3)+,R0                ;CONVERT OCTAL REGISTER CONTENTS
006272' 004577  171560          JSR     R5,@#BINASC             ;FOR 3 REGISTERS TO ASCII
006276' 000370                  .WORD   DVRDT1-.                ;AND PLACE IN THE MSG
006300' 012300                  MOV     (R3)+,R0
006302' 004577  171550          JSR     R5,@#BINASC
006306' 000376                  .WORD   DVRDT2-.
006310' 012300                  MOV     (R3)+,R0
006312' 004577  171540          JSR     R5,@#BINASC
006316' 000404                  .WORD   DVRDT3-.
006320' 012767  000050  000034  MOV     #40.,30$                ;INITIALIZE MSG LENGTH TO 3 REGS
006326' 162716  000003          SUB     #3,(SP)                 ;DECR REGISTER CNT
006332' 100005                  BPL     25$                     ;< 3 REGS? (Y,N-25$)
006334' 162767  000016  000020  20$:    SUB     #14.,30$        ;SHORTEN MSG LENGTH BY 1 REG
006342' 005216                  INC     (SP)                    ;INCR NEG REG CNT
006344' 100773                  BMI     20$                     ;CNT BACK TO 0? (Y,N-20$)
006346' 010346          25$:    MOV     R3,-(SP)                ;SAVE REG DATA PNTR
006350' 016603  000006          MOV     6(SP),R3                ;RESTORE PROG TBL ADR
006354' 004567  000060          JSR     R5,PRINT                ;GO PRINT THE MSG
006360' 000300                  .WORD   DVRGMG-.
006362' 000050          30$:    .WORD   40.
006364' 012603                  MOV     (SP)+,R3                ;RESTORE REG DATA PNTR
006366' 005716                  TST     (SP)                    ;MORE REGS TO GO?
006370' 001323                  BNE     10$                     ;N,Y-10$
006372' 005726                  TST     (SP)+                   ;REMOVE CNT FROM STACK
006374' 012605                  MOV     (SP)+,R5                ;RESTORE R5 & R3
006376' 012603                  MOV     (SP)+,R3
006400' 000205                  RTS     R5                      ;EXIT IN-LINE
```

```
                                        ;DISPLAY TRACK/SECT WORDS' VALUES

                                                ;JSR     PC,PRTIWD          S/R CALL
                                                ;DESTROYS R0,R1,R2

006402'                                 PRTIWD:
006402' 016700  171400                          MOV     HEAD,R0             ;GET & CONVERT TRACK VALUE
006406' 004577  171444                          JSR     R5,@BINASC
006412' 000771                                  .WORD   IFHEAD-.
006414' 016700  171370                          MOV     SECT,R0             ;GET & CONVERT SECT VALUE
006420' 004577  171432                          JSR     R5,@BINASC
006424' 000775                                  .WORD   IFSECT-.
006426' 004567  000006                          JSR     R5,PRINT            ;PRINT MSG WITH THEIR VALUES
006432' 000743                                  .WORD   INFOMG-.
006434' 000031                                  .WORD   25.
006436' 000207                                  RTS     PC                  ;EXIT IN-LINE


                                        ;ISSUE MSG TO LIST DEVICE SUBROUTINE

                                                ;JSR     R5,PRINT           S/R CALL
                                                ;.WORD   MSGADR-.           REL ADR OF MSG
                                                ;.WORD   BYTCNT             MSG BYTE CNT (IF NEGATIVE,
                                                ;                             RESET PRT DEV DEDICATED.)
                                                ;R3 = PROG TBL ADR
                                                ;R4 = FLAGWORD -- IF NEGATIVE, USE CMND MODE PRINT
                                                ;DESTROYS R0,R1,R2

006440' 010500                          PRINT:  MOV     R5,R0               ;GET MSG ADR & MAKE IT ABS
006442' 062500                                  ADD     (R5)+,R0
006444' 012501                                  MOV     (R5)+,R1            ;GET BYTE COUNT
006446' 005704                                  TST     R4                  ;USE CMND MODE PRINT?
006450' 100030                                  BPL     40$                 ;Y,N-40$
006452' 010702                                  MOV     PC,R2               ;SET UP LINK INFO ADR
006454' 062702  000040                          ADD     #20$-.,R2
006460' 160200                                  SUB     R2,R0               ;MAKE MSG ADR REL
006462' 010022                                  MOV     R0,(R2)+            ;STORE MSG ADR
006464' 010112                                  MOV     R1,(R2)             ;STORE MSG'S BYTE COUNT
006466' 100001                                  BPL     10$                 ;CNT NEG? (Y,N-10$)
006470' 005412                                  NEG     (R2)                ;MAKE IT POSITIVE
006472' 016367  000006  000056          10$:    MOV     PASCIN(R3),PROGNM   ;STORE PROG'S # IN MSG
006500' 004577  171350                          JSR     R5,@CLIST           ;ISSUE PROG #
006504' 000050                                  .WORD   PNMMSG-.
006506' 000005                                  .WORD   5
006510' 004577  171340                          JSR     R5,@CLIST           ;ISSUE MSG SPECIFIED
006514' 000000                          20$:    .WORD   XXXX
006516' 000000                                  .WORD   XXXX
006520' 004577  171330                          JSR     R5,@CLIST           ;ISSUE A <CR> & <LF>
006524' 000257                                  .WORD   CRLF-.
006526' 000002                                  .WORD   2
006530' 000410                                  BR      PRTEX               ;GO TO EXIT
006532' 010067  000010                  40$:    MOV     R0,50$              ;STORE MSG'S ABS ADR
006536' 010167  000006                          MOV     R1,60$              ;STORE ITS BYTE CNT
006542' 004577  171304                          JSR     R5,@ULIST           ;GO TO MPG TO ISSUE THE MSG
006546' 000000                          50$:    .WORD   XXXX
006550' 000000                          60$:    .WORD   XXXX
```

```
  1630  006552' 000205                PRTEX:  RTS     R5                      ;EXIT IN-LINE
```

```
                                        .SBTTL  RS03/RS04 MESSAGE STORAGE AREA


                                        .NLIST  BEX

                                        .EVEN
006554' 021520                  PNMMSG: .ASCII  /P#/
006556' 054130      011         PROGNM: .ASCII  /XX/<011>
006561'    101  020124  040514  ATIMSG: .ASCII  /AT LAST INT:/
006575'    103  051125  042522  CURMSG: .ASCII  /CURRENTLY:/
006607'    105  042116  047440  RENDMG: .ASCII  /END OF REPORT/<15><12>
                                        .EVEN
006626' 025052  025052  051522  UNITMG: .ASCII  /****RSXX DISK UNIT: /
006652' 054130  054130  054130  UNASCI: .ASCII  /XXXXXX/
                                        .EVEN
006660' 054130  054130  020075  DVRGMG: .ASCII  /XXXX= /
006666' 054130  054130  054130  DVRDT1: .ASCII  /XXXXXX  XXXX= /
006704' 054130  054130  054130  DVRDT2: .ASCII  /XXXXXX  XXXX= /
006722' 054130  054130  054130  DVRDT3: .ASCII  /XXXXXX/
006730' 054502  042524  035123  CNTSMG: .ASCII  /BYTES:  RD= /
006744' 054130  054130  054130  BCMRD:  .ASCII  /XXXXXXXXXXXX  WR = /
006767'    130  054130  054130  BCMWR:  .ASCII  /XXXXXXXXXXXX/
007003'    015      012         CRLF:   .ASCII  <015><012>
007005'    011  041411  036513          .ASCII  <011><011>/CK= /
007013'    130  054130  054130  BCMCK:  .ASCII  /XXXXXXXXXXXX/<015><012><011>/CMNDS:  RD= /
007046' 054130  054130  054130  CMDCRD: .ASCII  /XXXXXX WR = /
007062' 054130  054130  054130  CMDCWR: .ASCII  /XXXXXX CK  = /
007077'    130  054130  054130  CMDCCK: .ASCII  /XXXXXX/<015><012><011><011>/SK= /
007115'    130  054130  054130  CMDCSK: .ASCII  /XXXXXX CLR= /
007131'    130  054130  054130  CMDDRV: .ASCII  /XXXXXX DCLR= /
007146' 054130  054130  054130  CMDCMS: .ASCII  /XXXXXX/<015><012><011>/ERRORS: DEV= /
007174' 054130  054130  054130  CNTCEC: .ASCII  *XXXXXX DATA/OPR= *
007215'    130  054130  054130  CNTDER: .ASCII  /XXXXXX/<015><012><011>/RETRYS: DLT= /
007243'    130  054130  054130  CNTDLT: .ASCII  /XXXXXX DTE= /
007257'    130  054130  054130  CNTHCE: .ASCII  /XXXXXX DCK= /
007273'    130  054130  054130  CNTDCK: .ASCII  /XXXXXX/<015><012><011><011>/WCE= /
007312' 054130  054130  054130  CNTWCE: .ASCII  /XXXXXX/<015><012><011>/TOTAL RETRYS: /
007341'    130  054130  054130  CNTRTY: .ASCII  /XXXXXX/<015><012><011>/INTERRUPTS:  /
007367'    130  054130  054130  CNTINT: .ASCII  /XXXXXX/
           007375'                 CNTSEN= .
007375'    124  040522  036513  INFOMG: .ASCII  /TRAK= /
007403'    130  054130  054130  IFHEAD: .ASCII  /XXXXXX  SECT= /
007421'    130  054130  054130  IFSECT: .ASCII  /XXXXXX/
           007430'                 .EVEN
007430' 052123  047115  020124  STMNMG: .ASCII  /STMNT # /
007440' 054130  054130  054130  STMNUM: .ASCII  /XXXXXX/
007446' 051105  047522  020122  DKEMSG: .ASCII  /ERROR BITS:/<015><012><011>
007464' 000100                  CODFLD: .BLKB   64.
007564' 042502  047506  042522  BEFIO:  .ASCII  'BEFORE ISSUING I/O CMND:'
007614' 043101  042524  020122  AFTIO:  .ASCII  'AFTER ISSUING I/O CMND:'
007643'    124  047457  047440  CRTO:   .ASCIZ  'T/O ON CRESET'
007661'    124  046511  047505  IOTO:   .ASCIZ  'TIMEOUT ON I/O'
007700' 047516  026516  054105  NONEXD: .ASCIZ  'NON-EXISTENT DRIVE'
007723'    125  051516  043101  INITUS: .ASCII  'UNSAFE'
007732' 051105  047522  020122  INITDE: .ASCIZ  'ERROR ON INITIATION'
007756' 044504  045523  044440  OFFLIN: .ASCIZ  'DISK IS OFF-LINE'
```

M03


```
1688  007777'     116  047117  044455  NOITER: .ASCII  'NON-INT '
1689  010007'     111  047457  052040  IOTERM: .ASCIZ  'I/O TERMINATION ERROR'
1690  010035'     111  052116  053440  NOATA:  .ASCIZ  'INT WITHOUT ATA'
1691  010055'     125  042516  050130  UXPATA: .ASCIZ  'UNEXP ATA COND'
1692  010074' 054105  040510  051525  RTYEXH: .ASCIZ  'EXHAUSTED RETRIES'
1693  010116' 047111  020126  047125  INVDVN: .ASCIZ  /INV UNIT #/
1694          010132'                         .EVEN
1695                                          .LIST   BEX
1696
1697          010132'                 DVREND= .
```

```
                                .SBTTL  FORMATS FOR PROGRAM & DEVICE ROUTINE TABLES

                        ;       PROGRAM TABLE FORMAT

000242                  PTLGTH= 162.    ;PROGRAM TABLE LENGTH - NON MEM MGMNT VERSION OF MPG

                        ;(PTLGTH= 212.  ;PROGRAM TABLE LENGTH - MEM MGMNT VERSION OF MPG)

000000                  PFLGWD= +0.     ;PROGRAM FLAG WORD - 1 WORD

                                URSTOP= 2       ; 1 = USER HAS STOPPED THIS PROGRAM
                                ERSTOP= 4       ; 1 = AN ERROR HAS STOPPED THIS PROGRAM
                                WT4IOT= 10      ; 1 = WAITING FOR I/O TERMINATION
                                CTPRIO= 20      ; 1 = CONSOLE OR PRINTER I/O IN PROGRESS
                                SETDED= 40      ; 1 = THIS PROG SET THE PRT DEV DEDICATED FLAG
                                OCPRES= 100     ; 1 = OBJ CODE IS PRESENT
                                USEUBM= 200     ; 1 = THIS PROG USES THE UNIBUS MAP (MEM MGMNT ONLY)
                                ACTIVE= 100000  ; 1 = PROGRAM IS ACTIVE (SPECIFIED FOR EXECUTION)

000002                  POPSW=  +2.     ;PROGRAM'S OPERATION SWITCHES - 1 WORD

                                STONER= 100000  ; 1 = STOP PROG EXECUTION UPON ERROR
                                CYCPRG= 40000   ; 1 = CYCLE PROGRAM (ON CURRENT DEVICE)
                                PRONER= 20000   ; 1 = DO NOT PRINT ON ERROR
                                BIT12=  10000   ; 0 = NOT USED
                                BIT11=  4000    ; 0 = NOT USED
                                CYCDVL= 2000    ; 1 = CYCLE THE DEVICE LIST
                                GTNXTD= 1000    ; 1 = CYCLE ON SAME DEVICE UPON ERROR
                                DOERCK= 400     ; 1 = DON'T DO ERROR CHECKING
                                SPOPER= 200     ; 1 = DEVICE SPECIAL OPERATION
                                BIT6=   100     ; 0 = NOT USED
                                DOIOT=  40      ; 1 = DO NOT PERFORM I/O TIMEOUT
                                AUTORP= 20      ; 1 = DO NOT AUTOMATICALLY DISPLAY COUNTS
                                AURPEP= 10      ; 1 = AUTO DISPLAY COUNTS AT END OF FINAL PASS ONLY
                                HSKPEP= 4       ; 1 = HOUSEKEEP COUNTS ONLY AT RUN COMMAND
                                PFBBOV= 2       ; 1 = PRINT FIRST BAD BYTE ONLY ON VERIFY
                                NOCOMP= 1       ; 1 = DO NOT PRINT PROG COMPLETED MSG

000004                  PFWADR= +4.     ;*;PROGRAM FLAGWORD ADDRESS - 1 WORD

000006                  PASCIN= +6.     ;PROGRAM'S NUMBER IN ASCII - 1 WORD

000010                  PNAME=  +8.     ;PROGRAM'S NAME IN ASCII - 6 BYTES

000016                  PRDIOA= +14.    ;ADDRESS OF READ I/O AREA - 1 WORD

000020                  PWRIOA= +16.    ;ADDRESS OF WRITE I/O AREA - 1 WORD

000022                  PSRCST= +18.    ;SOURCE STATEMENTS START ADDRESS - 1 WORD

000024                  POBJST= +20.    ;OBJECT CODE START ADDRESS - 1 WORD

000026                  PLNGTH= +22.    ;PROG AREA LENGTH (OBJ END MINUS PROG TBL START) - 1 WORD

000030                  PTOCNT= +24.    ;I/O TIMEOUT COUNT - 1 WORD
```

```
1755            000032                  PMDLCD= +26.    ;DEV ROUT MODEL # CODE - 1 WORD
1756
1757            000034                  PDPNTR= +28.    ;CURRENT DEVICE NUMBER POINTER - 1 BYTE
1758
1759            000035                  PCURDV= +29.    ;CURRENT DEVICE # - 1 BYTE
1760
1761            000036                  PDNUMS= +30.    ;DEVICE NUMBERS - 16 BYTES
1762
1763            000056                  PTEM0=  +46.    ;USER PROGRAM TEMPORARY STORAGE - 1 WORD
1764
1765            000060                  PTEM1=  +48.    ;USER PROGRAM TEMPORARY STORAGE - 1 WORD
1766
1767            000062                  PTEM2=  +50.    ;USER PROGRAM TEMPORARY STORAGE - 1 WORD
1768
1769            000064                  PTEM3=  +52.    ;USER PROGRAM TEMPORARY STORAGE - 1 WORD
1770
1771            000066                  PTEM4=  +54.    ;USER PROGRAM TEMPORARY STORAGE - 1 WORD
1772
1773            000070                  PTEM5=  +56.    ;USER PROGRAM TEMPORARY STORAGE - 1 WORD
1774
1775            000072                  PTEM6=  +58.    ;USER PROGRAM TEMPORARY STORAGE - 1 WORD
1776
1777            000074                  PTEM7=  +60.    ;USER PROGRAM TEMPORARY STORAGE - 1 WORD
1778
1779            000076                  PTEM8=  +62.    ;USER PROGRAM TEMPORARY STORAGE - 1 WORD
1780
1781            000100                  PTEM9=  +64.    ;USER PROGRAM TEMPORARY STORAGE - 1 WORD
1782
1783            000102                  PTEM10= +66.    ;USER PROGRAM TEMPORARY STORAGE - 1 WORD
1784
1785            000104                  PTEM11= +68.    ;USER PROGRAM TEMPORARY STORAGE - 1 WORD
1786
1787            000106                  PTEM12= +70.    ;USER PROGRAM TEMPORARY STORAGE - 1 WORD
1788
1789            000110                  PTEM13= +72.    ;USER PROGRAM TEMPORARY STORAGE - 1 WORD
1790
1791            000112                  PTEM14= +74.    ;USER PROGRAM TEMPORARY STORAGE - 1 WORD
1792
1793            000114                  PTEM15= +76.    ;USER PROGRAM TEMPORARY STORAGE - 1 WORD
1794
1795            000116                  PNBR=   +78.    ;NUMBER OF BYTES TO TRANSFER ON MOVE (NBR) - 1 WORD
1796
1797            000120                  PSRC=   +80.    ;DATA SOURCE ADDRESS ON MOVE (SRC) - 1 WORD
1798
1799            000122                  PDST=   +82.    ;DATA DESTINATION ADDRESS ON MOVE (DST) - 1 WORD
1800
1801            000124                  PSTKCT= +84.    ;# OF WORDS (X 2) SAVED OFF STACK - 1 WORD
1802
1803            000126                  PSTKSV= +86.    ;STACK WORDS STORAGE AREA - 30 WORDS
1804
1805            000222                  PSVREG= +146.   ;USER'S R0 THRU R5 REGISTERS STORAGE AREA - 6 WORDS
1806
1807            000236                  PUSRPC= +158.   ;USER'S CURRENT PROGRAM COUNTER - 1 WORD
1808
```

```
1810                                    ;FOLLOWING ENTRIES (PRDIOX THRU PUBMAP) ARE ONLY IN MEM MGMNT VERSION
1811
1812                                ;(PRDIOX= +160. ;18/22 BIT ABSOLUTE ADDRESS OF READ I/O AREA - 2 WORDS)
1813
1814                                ;(PRDIOV= +164. ;18 BIT VIRTUAL ADDRESS OF READ I/O AREA - 2 WORDS)
1815
1816                                ;(PWRIOX= +168. ;18/22 BIT ABSOLUTE ADDRESS OF WRITE I/O AREA - 2 WORDS)
1817
1818                                ;(PWRIOV= +172. ;18 BIT VIRTUAL ADDRESS OF WRITE I/O AREA - 2 WORDS)
1819
1820                                ;(PUPARS= +176. ;STORAGE AREA FOR USER'S PAR'S 0 THRU 7 - 8 WORDS)
1821
1822                                ;(PUPDRS= +192. ;STORAGE AREA FOR USER'S PDR'S 0 THRU 7 - 8 WORDS)
1823
1824                                ;(PUBMAP= +208. ;1ST UNIBUS MAP REG # AND # OF REGS USED - 1 WORD)
1825
1826                                        ;END OF MEM MGMNT ONLY ENTRIES
1827
1828  000240                    PTSIZE= +160.   ;PROGRAM TABLE SIZE IN BYTES - 1 WORD - NON MEM MGMNT
1829
1830                                ;(PTSIZE= +210. ;PROGRAM TABLE SIZE IN BYTES - 1 WORD - MEM MGMNT VERSION)
1831
1832  000242                    PTEND=  +162.   ;END OF PROGRAM TABLE - NON MEM MGMNT VERSION
1833
1834                                ;(PTEND=  +212. ;END OF PROGRAM TABLE - MEM MGMNT VERSION)
```

```
                        ;       DEVICE ROUTINE TABLE
                        ;
000116                  DRTLTH= 78.     ;DEVICE ROUTINE TABLE LENGTH
                        ;
000000                  DEVRSZ= +0.     ;DEVICE ROUTINE SIZE IN BYTES - 1 WORD
000002                  DEVFWD= +2.     ;DEVICE ROUTINE FLAGWORD - 1 WORD
000004                  DEVIW1= +4.     ;DEVICE INTERFACE WORD # 1 - 1 WORD
000006                  DEVIW2= +6.     ;DEVICE INTERFACE WORD # 2 - 1 WORD
000010                  DEVIW3= +8.     ;DEVICE INTERFACE WORD # 3 - 1 WORD
000012                  DEVIW4= +10.    ;DEVICE INTERFACE WORD # 4 - 1 WORD
000014                  DEVIW5= +12.    ;DEVICE INTERFACE WORD # 5 - 1 WORD
000016                  DEVIW6= +14.    ;DEVICE INTERFACE WORD # 6 - 1 WORD
000020                  DEVIW7= +16.    ;DEVICE INTERFACE WORD # 7 - 1 WORD  (SIZE)
000022                  DEVIW8= +18.    ;DEVICE INTERFACE WORD # 8 - 1 WORD  (ERR)
000024                  DEVDRA= +20.    ;DEVICE REGISTERS ADDRESS - 1 WORD
000026                  DEVIVA= +22.    ;DEVICE INTERRUPT VECTOR ADDRESS - 1 WORD
000030                  DEVRPS= +24.    ;DEVICE READ PROCESSOR STATUS WORD (BUS REQ) - 1 WORD
000032                  DEVWPS= +26.    ;DEVICE WRITE PROC STATUS WORD (BUS REQ) - 1 WORD
000034                  DHKPAD= +28.    ;DEVICE ROUT HOUSEKEEPING ROUT REL ENTRY ADR - 1 WORD
000036                  DERPAD= +30.    ;DEVICE ROUT REPORT ROUT REL ENTRY ADR - 1 WORD
000040                  DKILAD= +32.    ;DEVICE ROUT KILL ROUTINE REL ENTRY ADR - 1 WORD
000042                  DECTAD= +34.    ;DEVICE ROUT ERROR COUNTER REL ADR - 1 WORD
000044                  DTOEAD= +36.    ;DEVICE ROUT TIMEOUT ERR ROUT REL ENTRY ADR - 1 WORD
000046                  DEVIOB= +38.    ;DEVICE I/O BUSY BRANCH ADDRESS (CIOBSY) - 1 WORD
000050                  DEVDER= +40.    ;DEVICE ERROR BRANCH ADDRESS (CUPGER) - 1 WORD
000052                  DVUPRT= +42.    ;USER MODE PRINT BRANCH ADDRESS (ULIST) - 1 WORD
000054                  DVCPRT= +44.    ;CMND MODE PRINT BRANCH ADDRESS (CLIST) - 1 WORD
000056                  DEVBTA= +46.    ;CONVERT BINARY TO ASCII BR ADR (BINASC) - 1 WORD
000060                  DV3TDA= +48.    ;CONVERT BINARY TO DECIMAL ASCII BR ADR (BTASLZ) - 1 WORD
```

```
1892            000062                  DVPDTA= +50.    ;CONVERT PACKED DECIMAL TO ASCII BR ADR (DECASC) - 1 WORD
1893
1894            000064                  DVSFWD= +52.    ;MPG SYSTEM FLAGWORD ADDRESS (CSYSFW) - 1 WORD
1895
1896            000066                  DVSVEC= +54.    ;SET INTERRUPT VECTOR BR ADR (SETVEC) - 1 WORD
1897
1898            000070                  DVCVEC= +56.    ;CLEAR INTERRUPT VECTOR BR ADR (CLRVEC) - 1 WORD
1899
1900            000072                  DVTVEC= +58.    ;TEST INTERRUPT VECTOR BR ADR (TSTVEC) - 1 WORD
1901
1902            000074                  DVRINT= +60.    ;RETURN FROM INTERRUPT BR ADR (RTNINT) - 1 WORD
1903
1904            000076                  DVGETB= +62.    ;GET DATA BYTE BR ADR (GETBYT) - 1 WORD
1905
1906            000100                  DVPUTB= +64.    ;PUT DATA BYTE BR ADR (PUTBYT) - 1 WORD
1907
1908            000102                  DEVSTP= +66.    ;DEVICE ROUT REL SYMBOL TABLE POINTER - 1 WORD
1909
1910            000104                  DEVETP= +68.    ;DEVICE ROUT REL ENTRY TABLE POINTER - 1 WORD
1911
1912            000106                  DVPTEP= +70.    ;PACK TABLE EXTEN. REL POINTER - 1 WORD
1913
1914            000110                  DVVTEP= +72.    ;VECTOR TABLE EXTEN. REL POINTER - 1 WORD
1915
1916            000112                  DVCTEP= +74.    ;COMPILER TBL EXTEN. REL POINTER - 1 WORD
1917
1918            000114                  DVIWSP= +76.    ;DEVICE INTERFACE WORD SYMBOL TBL REL POINTER - 1 WORD
1919
1920            000116                  DRTEND= +78.    ;END OF DEVICE ROUTINE TABLE
1921
1922
1923            000001                          .END
```

```
ACQEAD  003056R  002    CNTDLT  007243R  002    DEVIW8= 000022           EBSTAT  005016R  002    MSFMT5  001072R  002
ACQERC  003060R  002    CNTEND= 001234R  002    DEVRPS= 000030           ERCDTB  005312R  002    MYATA   003120R  002
ACQERR  003046R  002    CNTHCE  007257R  002    DEVRSZ= 000000           ERMBAS  004762R  002    NED   = 010000
ACQGDK  003070R  002    CNTINT  007367R  002    DEVSTP= 000102           ERR   = 040000           NOATA   010035R  002
ACQHSK  002462R  002    CNTNUM= 000024           DEVWPS= 000032           ERRADR  001234R  002    NOCOMP= 000001
ACQRTY  002464R  002    CNTRTY  007341R  002    DFLGWD  000002R  002    ERRCNT  001214R  002    NOITER  007777R  002
ACTIVE= 100000           CNTSEN= 007375R  002    DHKPAD= 000034           ERRCOM  004622R  002    NONEXD  007700R  002
AFTIO   007614R  002    CNTSMG  006730R  002    DISCNT  001564R  002    ERRCS   004562R  002    NOWAIT  002212R  002
ANYIOI= 000200           CNTWCE  007312R  002    DISPST  006202R  002    ERRCS1  004570R  002    NRSEC = 000100
APORT   002222R  002    CODFLD  007464R  002    DKEMSG  007446R  002    ERREX   005310R  002    NRTRK = 000100
ATA   = 100000           CORFLG= 002000           DKILAD= 000040           ERRFND  003756R  002    OCPRES= 000100
ATATBL  003100R  002    COUNTS  001164R  002    DLTCNT  001220R  002    ERRI    000022R  002    ODD     002242R  002
ATIMSG  006561R  002    CPU70 = 000010           DOERCK= 000400           ERRIS   004606R  002    OFFLIN  007756R  002
AURPEP= 000010           CRESET  002302R  002    DOIOT = 000040           ERRSNM  005230R  002    OTHATA  003122R  002
AUTORP= 000020           CRLF    007003R  002    DOTERM= 000002           ERSTAD  005222R  002    PARITY= 000020
BAI   = 000010           CRTO    007643R  002    DPR   = 000400           ERSTOP= 000004           PASCIN= 000006
BAIOFF  002272R  002    CSTAT   001130R  002    DREGAD  000024R  002    EVEN    002252R  002    PATCH   001264R  002
BAION   002262R  002    CSYSFW  000064R  002    DRESET  002352R  002    FINCNT  001252R  002    PC    =%000007
BCMCK   007013R  002    CTLCNT  001210R  002    DRTEND= 000116           GETBYT  000076R  002    PCURDV= 000035
BCMRD   006744R  002    CTPRIO= 000020           DRTLTH= 000116           GO    = 000001           PDNUMS= 000036
BCMWR   006767R  002    CUPGER  000050R  002    DRVCNT  001212R  002    GOTDSK  002630R  002    PDPNTR= 000034
BEFIO   007564R  002    CURADR  001244R  002    DTECNT  001222R  002    GTNXTD= 001000           PDST  = 000122
BINASC  000056R  002    CURCMD  001242R  002    DTOEAD= 000044           HARDER  004176R  002    PFBBOV= 000002
BIT11 = 004000           CURCNT  001250R  002    DVA   = 004000           HEAD    000006R  002    PFLGWD= 000000
BIT12 = 010000           CURFLG  001240R  002    DVBTDA= 000060           HSKEEP  001334R  002    PFWADR= 000004
BIT6  = 000100           CURMSG  006575R  002    DVCMDS  000242R  002    HSKPEN= 001260R  002    PLNGTH= 000026
BPORT   002232R  002    CURRTY  001254R  002    DVCPRT= 000054           HSKPEP= 000004           PMDLCD= 000032
BTASLZ  000060R  002    CYCDVL= 002000           DVCPTE  000654R  002    HSKPST= 001074R  002    PNAME = 000010
BYCK    001174R  002    CYCPRG= 040000           DVCTEP= 000112           IE    = 000100           PNBR  = 000116
BYRD    001164R  002    DATAER  001216R  002    DVCVEC= 000070           IFHEAD  007403R  002    PNMMSG  006554R  002
BYWR    001170R  002    DCCODE= 000010           DVGETB= 000076           IFSECT  007421R  002    POBJST= 000024
CIOBSY  000046R  002    DCKCNT  001224R  002    DVIWSP= 000114           INFOMG  007375R  002    POPSW = 000002
CKCNT   001204R  002    DECASC  000062R  002    DVIWST  001032R  002    INITDE  007732R  002    PORT  = 002000
CKCORR  004216R  002    DECTAD= 000042           DVMVTE  000554R  002    INITUS  007723R  002    PRDIOA= 000016
CKDBSY  004230R  002    DERPAD= 000036           DVPDTA= 000062           INTCNT  001232R  002    PRINT   006440R  002
CKRTRY  004016R  002    DEVBTA= 000056           DVPKTE  000354R  002    INTEAD  004454R  002    PROCEX  004474R  002
CKSC    003654R  002    DEVDER= 000050           DVPTEP= 000106           INTEX   003644R  002    PROCTM  004360R  002
CLIST   000054R  002    DEVDRA= 000024           DVPUTB= 000100           INVDVN  010116R  002    PROGNM  006556R  002
CLR   = 000040           DEVETP= 000104           DVRDT1  006666R  002    IOERR = 000001           PRONER= 020000
CLRVEC  000070R  002    DEVFWD= 000002           DVRDT2  006704R  002    IOTERM  010007R  002    PRTEX   006552R  002
CLRWTF  003622R  002    DEVID   006066R  002    DVRDT3  006722R  002    IOTO    007661R  002    PRTIWD  006402R  002
CMDCCK  007077R  002    DEVIIV  006150R  002    DVREGE= 000242R  002    ISTAT = 001074R  002    PS    = 177776
CMDCMS  007146R  002    DEVIML  006176R  002    DVREGS  000116R  002    ITIME = 072460           PSEL  = 002000
CMDCOM  003256R  002    DEVIOB= 000046           DVREND= 010132R  002    IVCTAD  000026R  002    PSRC  = 000120
CMDCRD  007046R  002    DEVIPR  006170R  002    DVREX   001664R  002    JSETER  004212R  002    PSRCST= 000022
CMDCSK  007115R  002    DEVIVA= 000026           DVRGMG  006660R  002    KILL    002024R  002    PSTKCT= 000124
CMDCWR  007062R  002    DEVIW1= 000004           DVRINT= 000074           KILLEX  002054R  002    PSTKSV= 000126
CMDDRV  007131R  002    DEVIW2= 000006           DVSFWD= 000064           LOCZ    000000R  002    PSVREG= 000222
CMDISU= 000100           DEVIW3= 000010           DVSVEC= 000066           MCPE  = 020000           PSWD    000030R  002
CNTADR  001236R  002    DEVIW4= 000012           DVTVEC= 000072           MMVER = 000001           PTEM0 = 000056
CNTCEC  007174R  002    DEVIW5= 000014           DVUPRT= 000052           MOL   = 010000           PTEM1 = 000060
CNTDCK  007273R  002    DEVIW6= 000016           DVVTEP= 000110           MSFMT1  001064R  002    PTEM10= 000102
CNTDER  007215R  002    DEVIW7= 000020           EBSBAS  005014R  002    MSFMT2  001065R  002    PTEM11= 000104
```

```
PTEM12= 000106            RENDMG  006607R  002   RPMR   = 000024          SKCNT   001206R  002   TVECTX  004560R  002
PTEM13= 000110            REPORT  001406R  002   RPTBAS  001630R  002    SP    =%000006          ULIST   000052R  002
PTEM14= 000112            REPTBL  001674R  002   RPTEND  001654R  002    SPOPER= 000200          UNASCI  006652R  002
PTEM15= 000114            RESREG  005720R  002   RPTLP   001612R  002    STEPDN  002142R  002   UNITMG  006626R  002
PTEM2  = 000062           RETRYS  001230R  002   RPWC   = 000002          STEPD1  002162R  002   UNS    = 040000
PTEM3  = 000064           RHPINT  003546R  002   RTNINT  000074R  002    STEPD2  002150R  002   URSTOP= 000002
PTEM4  = 000066           RINTEX  004530R  002   RTRY    000012R  002    STEPEX  002106R  002   USEUBM= 000200
PTEM5  = 000070           RINTV   004506R  002   RTRYIP  001256R  002    STEPUP  002056R  002   USMTPS= 000002
PTEM6  = 000072           RPAS   = 000016         RTYEXH  010074R  002    STEPU1  002076R  002   UXPATA  010055R  002
PTEM7  = 000074           RPBA   = 000004         R0     =%000000         STEPU2  002064R  002   WAIT    002172R  002
PTEM8  = 000076           RPBAE  = 000030         R1     =%000001         STMNMG  007430R  002   WAITMD= 100000
PTEM9  = 000100           RPCS1  = 000000         R2     =%000002         STMNUM  007440R  002   WCCODE= 000050
PTEND  = 000242           RPCS1V  001260R  002   R3     =%000003         STONER= 100000          WCECNT  001226R  002
PTLGTH= 000242            RPCS2  = 000010         R4     =%000004         STPCOM  002120R  002   WCODE  = 000060
PTOCNT= 000030            RPCS2V  001262R  002   R5     =%000005         STSLUP= 001011          WRCK    003204R  002
PTSIZE= 000240            RPCS3  = 000032         SAVREG  005704R  002    STSTAT  005756R  002   WRCNT   001202R  002
PUSRPC= 000236            RPDA   = 000006         SCODE  = 000030         SUIORG  003440R  002   WRITE   003154R  002
PUTBYT  000100R     002   RPDB   = 000022         SEARCH  003234R  002    SUPTAD  005736R  002   WT4IOT= 000010
PWRIOA= 000020            RPDS   = 000012         SECT    000010R  002    TOUTER  001744R  002   XXXX   = 000000
RCODE  = 000070           RPDT   = 000026         SETDED= 000040          TRAK   = 000006R  002   .      = 010132R  002

. ABS.  000000     000
        000000     001
RJP11   010132     002

 ERRORS DETECTED:  0
 DEFAULT GLOBALS GENERATED:  0

*,DTRSAA/NL:TOC/DOC=DTRSAA.P11
RUN-TIME: 5 12 1 SECONDS
RUN-TIME RATIO: 24/19=1.2
CORE USED:  5K  (9 PAGES)

DOCUMENT PAGES:  45
```

Spooler runtime 7 Seconds, 32 KCS, 178 disk reads, 3 disk writes, 45 pages