```
.NLIST
.NLIST  SEQ,LO,BIN                                                  SEQ 0001
.LIST
.REM    %
```

IDENTIFICATION

PRODUCT CODE: AC-F087D-MC

PRODUCT NAME: CXRPDD0 RH70/RP04,5,6 MODULE

PRODUCT DATE: JANUARY 1980

MAINTAINER: DEC/X11 SUPPORT GROUP

1. ABSTRACT

RPD IS AN IOMODX WHICH WILL EXERSIZE UP TO 8 RP04/5/6 DISK DRIVES ON AN RH11/RH70 CONTROLLER, OR DUAL-PORTED BETWEEN TWO SUCH CONTROLLERS. IN SINGLE-PORT MODE, THE ENTIRE SURFACE OF EACH DISK WILL BE TESTABLE (RANDOMLY OR SYSTEMATICALLY).

IN DUAL-PORT MODE, THE "A" CONTROLLER MODULE WILL ACCESS ONLY THE LOW HALF OF THE DISK, AND THE "B" CONTROLLER MODULE WILL ACCESS ONLY THE HIGH HALF OF THE DISK; NO COMMUNICATION WILL BE ATTEMPTED BETWEEN THE TWO PORTS. ONE COPY OF THIS MODULE MUST BE CONFIGURED FOR EACH PORT.

SINGLE/DUAL PORT MODE WILL BE AUTOMATICALLY SELECTED ON THE BASIS OF THE RPDT REGISTER OF EACH SELECTED DRIVE.

DURING TESTING, ANY BAD SECTOR FOUND WILL BE STORED AUTOMATICALLY AND WITHOUT OPERATOR INTERVENTION INTO THE MODULE'S BADSPOT TABLE. A MESSAGE CAN BE PRINTED TO NOTIFY THE OPERATOR OF THE BADSPOT ADDRESS (SEE OPERATOR OPTIONS). SECTORS LISTED IN THE TABLE WILL NOT BE TESTED ON ANY DRIVE.

THE WBUFRQ WORD IN THE HEADER MAY BE CHANGED AT WILL TO ALTER THE MODULE'S TRANSFER SIZE (AND OF COURSE A AND B-PORT SIZES NEED NOT MATCH): THIS WILL NOT ALTER THE EFFECTIVENESS OF THE BADSPOT TABLE OR CAUSE THE MODULE TO ATTEMPT ACCESS OF OUT-OF-RANGE DISK ADDRESSES. WITHIN THIS FRAMEWORK, THE MODULES WILL ACCESS AS MUCH OF THE DISK AS POSSIBLE. RANDOM SEEKS WILL BE USED UNLESS THIS OPTION IS DESELECTED IN SR1.

2. REQUIREMENTS

HARDWARE: 1 TO 8 RP04/5/6 DRIVES ON 1 OR 2 RH11/RH70 CONTROLLERS

STORAGE:: RPD REQUIRES:

DECIMAL WORDS:	1806
OCTAL WORDS:	3416
OCTAL BYTES:	7034

3. PASS DEFINITION

- ONE PASS CONSISTS OF 300 ITERATIONS.
- AN ITERATION CONSISTS OF THE FOLLOWING STEPS EXECUTED ON EACH SELECTED DRIVE:
  - A. WRITE DATA TO DISK. IN DUAL-PORT MODE, PORT "A" MODULE WILL TEST THE LOW HALF OF THE DISK (CYLINDERS 0 TO 407 FOR RP06, CYLINDERS 0 TO 205 FOR RP04/5), PORT "B" WILL EXERSIZE THE HIGH HALF (CYLINDERS 206 TO 410 FOR RP04/5, CYLINDERS 408 TO 814 FOR RP06). IN SINGLE-PORT MODE, THE MODULE WILL ACCESS THE ENTIRE DISK SURFACE (CYLINDERS 0 TO 410 FOR RP04/5, CYLINDERS 0 TO 814 FOR RP06).
  - B. WRITE-CHECK DATA JUST WRITTEN.
  - C. READ 256 WORDS (1 SECTOR) INTO MODULE READ BUFFER.
  - D. DO IN-CORE COMPARE OF READ BUFFER WITH FIRST 256 WORDS OF WRITE BUFFER.
  - E. RELEASE THE DRIVE FOR OTHER PORT TO TEST.

4. EXECUTION TIME
--------- ----

ONE PASS OF RPD RUNNING ALONE ON A PDP-11/70 TAKES ABOUT A MINUTE.

5. CONFIGURATION REQUIREMENTS
------------- ------------

DEFAULT PARAMETERS:
DVA-176700 VCT-254 BR1-5 DVC-1

REQUIRED PARAMETERS:
SR1 MUST BE SET UP TO INDICATE WHICH PORT (A/B) THIS COPY OF THE MODULE IS TO TEST, IF ANY DRIVES ARE DUAL-PORTED. ALSO, BIT 7 MUST BE SET IF THE CONTROLLER IS AN RH70, AND CLEARED IF IT IS NOT AN RH70.

6. DEVICE/OPTION SETUP
------------- -----

MAKE SURE ALL DRIVES ARE POWERED UP, WRITE ENABLED, AND READY.
FOR DUAL-PORTED DRIVES: MAKE SURE CONTROLLER SELECT SWITCH IS IN A/B POSITION. IF DRIVE IS NOT CYCLED UP WITH SWITCH IN CORRECT POSITION, PLACE SWITCH CORRECTLY, AND DISABLE THEN RE-ENABLE DRIVE USING DRIVE DISABLE SWITCH.

7. OPERATOR OPTIONS
-------- -------

SR1

BIT 2
CLEAR(0):
TYPE OUT DATA LATE ERRORS (HARD ERROR) AND COUNT THEM INTERNALLY.
SET(1):
COUNT DATA LATE ERRORS INTERNALLY, BUT DO NOT TYPE OUT.

BIT 4
CLEAR(0):
THIS COPY OF RPD TESTS A-PORT ON ANY DUAL-PORTED DRIVES
SET(1):
THIS COPY OF RPD TESTS B-PORT ON ANY DUAL-PORTED DRIVES

BIT 5
CLEAR(0):
TEST DISKS WITH RANDOM SEEKS
SET(1):
DISABLE RANDOM SEEKS. INCREMENT SECTOR BY ONE EACH CYCLE.

BIT 6

CLEAR(0):
DO NOT TYPE OUT BADSPOTS.
SET(1):
ALWAYS TYPE OUT ANY BADSPOT NOT ALREADY RECORDED ON BADSPOT TABLE.

BIT 7
CLEAR(0):
CONTROLLER IS RH11
SET(1):
CONTROLLER IS RH70 (HAS RHBAE AND RHCS3)

8. NON-STANDARD PRINTOUTS

A. MOST PRINTOUTS HAVE THE STANDARD FORMATS.

B. ERROR MESSAGE DUMP RH11 REGISTERS IN THE FOLLOWING ORDER:

RHCS1	RHWC	RHBA	RPDA	RHCS2	RPDS	RPER1	RPAS
RPLA	RHDB	RPMR	RPDT	RPSN	RPOF	RPDC	RPCC
RPER2	RPER3	RPEC1	RPEC2				

ADDITIONALLY FOR THE RH70 RPBAE RHCS3

C. THE BAD SECTOR MESSAGE (WHEN ENABLED) LOOKS LIKE:
'DRIVE X: BADSPOT AT (OCTAL) CYL: XX, TRK: XX, SEC: XX'
WHERE 'X' IS AN OCIAL DIGIT. NO ERROR IS ASSOCIATED WITH THIS MESSAGE.

10. CHANGE HISTORY

NOTE: HISTORY STARTS WITH REV. D0

D0: ENHANCED ERROR REPORTING ROUTINES.

%

```
   369                                  .SBTTL  MODULE HEADER BLOCK                                                    ;DRB001
   370  000000'                                 IOMODX <RPDD  >,176700,254,5,0,0,1500,104,RBUF,256.,256.
   (1)  000000'                                 MODULE  150000,RPDD  ,176700,254,5,0,0,1500,104,RBUF,256.,256.
   (2)                                          .TITLE  RPDD DEC/X11 SYSTEM EXERCISER MODULE
   (2)                                  ;       DDXCOM  VERSION  6      23-MAY-78
   (2)                                          .LIST   BIN
   (2)                                  ;*****************************************************************************
   (2)  000000'                         BEGIN:
   (2)  000000' 050122  042104     040 MODNAM: .ASCII  /RPDD / ;MODULE NAME.
   (2)  000005'    000                  XFLAG:  .BYTE   OPEN                ;USED TO KEEP TRACK OF WBUFF USAGE
   (2)  000006' 176700                  ADDR:   176700+0                    ;1ST DEVICE ADDR.
   (2)  000010' 000254                  VECTOR: 254+0                       ;1ST DEVICE VECTOR.
   (2)  000012'    240                  BR1:    .BYTE   PRTY5+0             ;1ST BR LEVEL.
   (2)  000013'    000                  BR2:    .BYTE   PRTY0+0             ;2ND BR LEVEL.
   (2)  000014' 000001                  DVID1:  0+1                         ;DEVICE INDICATOR 1.
   (2)  000016' 000000                  SR1:    OPEN                        ;SWITCH REGISTER 1
   (2)  000020' 000000                  SR2:    OPEN                        ;SWITCH REGISTER 2
   (2)  000022' 000000                  SR3:    OPEN                        ;SWITCH REGISTER 3
   (2)  000024' 000000                  SR4:    OPEN                        ;SWITCH REGISTER 4
   (2)                                  ;*****************************************************************************
   (2)  000026' 150000                  STAT:   150000                      ;STATUS WORD.
   (2)  000030' 002054'                 INIT:   START                       ;MODULE START ADDR.
   (2)  000032' 000252'                 SPOINT: MODSP                       ;MODULE STACK POINTER.
   (2)  000034' 000000                  PASCNT: 0                           ;PASS COUNTER.
   (2)  000036' 001500                  ICONT:  1500                        ;# OF ITERATIONS PER PASS=1500
   (2)  000040' 000000                  ICOUNT: 0                           ;LOC TO COUNT ITERATIONS
   (2)  000042' 000000                  SOFCNT: 0                           ;LOC TO SAVE TOTAL SOFT ERRORS
   (2)  000044' 000000                  HRDCNT: 0                           ;LOC TO SAVE TOTAL HARD ERRORS
   (2)  000046' 000000                  SOFPAS: 0                           ;LOC TO SAVE SOFT ERRORS PER PASS
   (2)  000050' 000000                  HRDPAS: 0                           ;LOC TO SAVE HARD ERRORS PER PASS
   (2)  000052' 000000                  SYSCNT: 0                           ;# OF SYS ERRORS ACCUMULATED
   (2)  000054' 000000                  RANNUM: 0                           ;HOLDS RANDOM # WHEN RAND MACRO IS CALLED
   (2)  000056'                         CONFIG:                             ;RESERVED FOR MONITOR USE
   (2)  000056' 000000                  RES1:   0                           ;RESERVED FOR MONITOR USE
   (2)  000060' 000000                  RES2:   0                           ;RESERVED FOR MONITOR USE
   (2)  000062' 000000                  SVR0:   OPEN                        ;LOC TO SAVE R0.
   (2)  000064' 000000                  SVR1:   OPEN                        ;LOC TO SAVE R1.
   (2)  000066' 000000                  SVR2:   OPEN                        ;LOC TO SAVE R2.
   (2)  000070' 000000                  SVR3:   OPEN                        ;LOC TO SAVE R3.
   (2)  000072' 000000                  SVR4:   OPEN                        ;LOC TO SAVE R4.
   (2)  000074' 000000                  SVR5:   OPEN                        ;LOC TO SAVE R5.
   (2)  000076' 000000                  SVR6:   OPEN                        ;LOC TO SAVE R6.
   (2)  000100' 000000                  CSRA:   OPEN                        ;ADDR OF CURRENT CSR.
   (2)  000102'                         SBADR:                              ;ADDR OF GOOD DATA, OR
   (2)  000102' 000000                  ACSR:   OPEN                        ;CONTENTS OF CSR.
   (2)  000104'                         WASADR:                             ;ADDR OF BAD DATA, OR
   (2)  000104' 000000                  ASTAT:  OPEN                        ;STATUS REG CONTENTS.
   (2)  000106'                         ERRTYP:                             ;TYPE OF ERROR
   (2)  000106' 000000                  ASB:    OPEN                        ;EXPECTED DATA.
   (2)  000110' 000000                  AWAS:   OPEN                        ;ACTUAL DATA.
   (2)  000112' 002236'                 RSTRT:  RESTRT                      ;RESTART ADDRESS AFTER END OF PASS
   (2)  000114' 000000                  WDTO:   OPEN                        ;WORDS TO MEMORY PER ITERATION
   (2)  000116' 000000                  WDFR:   OPEN                        ;WORDS FROM MEMORY PER ITERATION
   (2)  000120' 000000                  INTR:   OPEN                        ;# OF INTERRUPTS PER ITERATION
   (2)  000122' 000104                  IDNUM:  104                         ;MODULE IDENTIFICATION NUMBER=104
   (2)  000124' 000722'                 RBUFVA: RBUF                        ;READ BUFFER VIRTUAL ADDRESS
```



```
   (2)  000126' 000000                  RBUFPA: OPEN                        ;READ BUFFER PHYSICAL ADDRESS
   (2)  000130' 000000                  RBUFEA: OPEN                        ;READ BUFFER EA BITS
   (2)  000132' 000400                  RBUFSZ: 256.                        ;SIZE OF THE READ BUFFER
   (2)  000134' 000000                  WBUFPA: OPEN                        ;WRITE BUFFER PHYSICAL ADDRESS
   (2)  000136' 000000                  WBUFEA: OPEN                        ;WRITE BUFFER EA BITS
   (2)  000140' 000400                  WBUFRQ: 256.                        ;WRITE BUFFER SIZE REQUESTED
   (2)  000142' 000000                  WBUFSZ: OPEN                        ;WRITE BUFFER SIZE AVAILABLE
   (2)  000144' 000000                  CDERCT: OPEN                        ;CDATA/DATCK ERROR COUNT
   (2)  000146' 000000                  CDWDCT: OPEN                        ;CDATA/DATCK WORD COUNT
   (2)  000150' 000000                  FREE:   OPEN                        ;RESERVED FOR FUTURE USE
   (2)          000040                          .REPT   SPSIZ               ;MODULE STACK STARTS HERE.
   (2)                                          .NLIST
   (2)                                          .WORD   0
   (2)                                          .LIST
   (3)                                          .ENDR
   (2)  000252'                         MODSP:
   (2)                                  ;*****************************************************************************
```

```
372                                     .SBTTL  MODULE PRIVATE DATA                                     ;DRB001
373  000252'                            BADSPT:                 ;BAD SPOT TABLE.  THERE IS ROOM FOR 48  ;DRB001
374                                                             ;BAD SPOTS, IN TWO-WORD FORMAT, CYLINDER ;DRB001
375                                                             ;FIRST, THEN SECTOR IN LOW BYTE OF SECOND ;DRB001
376                                                             ;WORD, AND TRACK IN HIGH BYTE.          ;DRB001
380  000252' 177777                             177777          ;CYLINDER ADDRESS                       ;DRB001
(1)  000254' 177777                             177777          ;TRACK/SECTOR ADDRESS                   ;DRB001
(1)  000256' 177777                             177777          ;CYLINDER ADDRESS                       ;DRB001
(1)  000260' 177777                             177777          ;TRACK/SECTOR ADDRESS                   ;DRB001
(1)  000262' 177777                             177777          ;CYLINDER ADDRESS                       ;DRB001
(1)  000264' 177777                             177777          ;TRACK/SECTOR ADDRESS                   ;DRB001
(1)  000266' 177777                             177777          ;CYLINDER ADDRESS                       ;DRB001
(1)  000270' 177777                             177777          ;TRACK/SECTOR ADDRESS                   ;DRB001
(1)  000272' 177777                             177777          ;CYLINDER ADDRESS                       ;DRB001
(1)  000274' 177777                             177777          ;TRACK/SECTOR ADDRESS                   ;DRB001
(1)  000276' 177777                             177777          ;CYLINDER ADDRESS                       ;DRB001
(1)  000300' 177777                             177777          ;TRACK/SECTOR ADDRESS                   ;DRB001
(1)  000302' 177777                             177777          ;CYLINDER ADDRESS                       ;DRB001
(1)  000304' 177777                             177777          ;TRACK/SECTOR ADDRESS                   ;DRB001
(1)  000306' 177777                             177777          ;CYLINDER ADDRESS                       ;DRB001
(1)  000310' 177777                             177777          ;TRACK/SECTOR ADDRESS                   ;DRB001
(1)  000312' 177777                             177777          ;CYLINDER ADDRESS                       ;DRB001
(1)  000314' 177777                             177777          ;TRACK/SECTOR ADDRESS                   ;DRB001
(1)  000316' 177777                             177777          ;CYLINDER ADDRESS                       ;DRB001
(1)  000320' 177777                             177777          ;TRACK/SECTOR ADDRESS                   ;DRB001
(1)  000322' 177777                             177777          ;CYLINDER ADDRESS                       ;DRB001
(1)  000324' 177777                             177777          ;TRACK/SECTOR ADDRESS                   ;DRB001
(1)  000326' 177777                             177777          ;CYLINDER ADDRESS                       ;DRB001
(1)  000330' 177777                             177777          ;TRACK/SECTOR ADDRESS                   ;DRB001
(1)  000332' 177777                             177777          ;CYLINDER ADDRESS                       ;DRB001
(1)  000334' 177777                             177777          ;TRACK/SECTOR ADDRESS                   ;DRB001
(1)  000336' 177777                             177777          ;CYLINDER ADDRESS                       ;DRB001
(1)  000340' 177777                             177777          ;TRACK/SECTOR ADDRESS                   ;DRB001
(1)  000342' 177777                             177777          ;CYLINDER ADDRESS                       ;DRB001
(1)  000344' 177777                             177777          ;TRACK/SECTOR ADDRESS                   ;DRB001
(1)  000346' 177777                             177777          ;CYLINDER ADDRESS                       ;DRB001
(1)  000350' 177777                             177777          ;TRACK/SECTOR ADDRESS                   ;DRB001
(1)  000352' 177777                             177777          ;CYLINDER ADDRESS                       ;DRB001
(1)  000354' 177777                             177777          ;TRACK/SECTOR ADDRESS                   ;DRB001
(1)  000356' 177777                             177777          ;CYLINDER ADDRESS                       ;DRB001
(1)  000360' 177777                             177777          ;TRACK/SECTOR ADDRESS                   ;DRB001
(1)  000362' 177777                             177777          ;CYLINDER ADDRESS                       ;DRB001
(1)  000364' 177777                             177777          ;TRACK/SECTOR ADDRESS                   ;DRB001
(1)  000366' 177777                             177777          ;CYLINDER ADDRESS                       ;DRB001
(1)  000370' 177777                             177777          ;TRACK/SECTOR ADDRESS                   ;DRB001
(1)  000372' 177777                             177777          ;CYLINDER ADDRESS                       ;DRB001
(1)  000374' 177777                             177777          ;TRACK/SECTOR ADDRESS                   ;DRB001
(1)  000376' 177777                             177777          ;CYLINDER ADDRESS                       ;DRB001
(1)  000400' 177777                             177777          ;TRACK/SECTOR ADDRESS                   ;DRB001
(1)  000402' 177777                             177777          ;CYLINDER ADDRESS                       ;DRB001
(1)  000404' 177777                             177777          ;TRACK/SECTOR ADDRESS                   ;DRB001
(1)  000406' 177777                             177777          ;CYLINDER ADDRESS                       ;DRB001
(1)  000410' 177777                             177777          ;TRACK/SECTOR ADDRESS                   ;DRB001
(1)  000412' 177777                             177777          ;CYLINDER ADDRESS                       ;DRB001
(1)  000414' 177777                             177777          ;TRACK/SECTOR ADDRESS                   ;DRB001
(1)  000416' 177777                             177777          ;CYLINDER ADDRESS                       ;DRB001
```



```
(1)  000420' 177777                             177777          ;TRACK/SECTOR ADDRESS                   ;DRB001
(1)  000422' 177777                             177777          ;CYLINDER ADDRESS                       ;DRB001
(1)  000424' 177777                             177777          ;TRACK/SECTOR ADDRESS                   ;DRB001
(1)  000426' 177777                             177777          ;CYLINDER ADDRESS                       ;DRB001
(1)  000430' 177777                             177777          ;TRACK/SECTOR ADDRESS                   ;DRB001
(1)  000432' 177777                             177777          ;CYLINDER ADDRESS                       ;DRB001
(1)  000434' 177777                             177777          ;TRACK/SECTOR ADDRESS                   ;DRB001
(1)  000436' 177777                             177777          ;CYLINDER ADDRESS                       ;DRB001
(1)  000440' 177777                             177777          ;TRACK/SECTOR ADDRESS                   ;DRB001
(1)  000442' 177777                             177777          ;CYLINDER ADDRESS                       ;DRB001
(1)  000444' 177777                             177777          ;TRACK/SECTOR ADDRESS                   ;DRB001
(1)  000446' 177777                             177777          ;CYLINDER ADDRESS                       ;DRB001
(1)  000450' 177777                             177777          ;TRACK/SECTOR ADDRESS                   ;DRB001
(1)  000452' 177777                             177777          ;CYLINDER ADDRESS                       ;DRB001
(1)  000454' 177777                             177777          ;TRACK/SECTOR ADDRESS                   ;DRB001
(1)  000456' 177777                             177777          ;CYLINDER ADDRESS                       ;DRB001
(1)  000460' 177777                             177777          ;TRACK/SECTOR ADDRESS                   ;DRB001
(1)  000462' 177777                             177777          ;CYLINDER ADDRESS                       ;DRB001
(1)  000464' 177777                             177777          ;TRACK/SECTOR ADDRESS                   ;DRB001
(1)  000466' 177777                             177777          ;CYLINDER ADDRESS                       ;DRB001
(1)  000470' 177777                             177777          ;TRACK/SECTOR ADDRESS                   ;DRB001
(1)  000472' 177777                             177777          ;CYLINDER ADDRESS                       ;DRB001
(1)  000474' 177777                             177777          ;TRACK/SECTOR ADDRESS                   ;DRB001
(1)  000476' 177777                             177777          ;CYLINDER ADDRESS                       ;DRB001
(1)  000500' 177777                             177777          ;TRACK/SECTOR ADDRESS                   ;DRB001
(1)  000502' 177777                             177777          ;CYLINDER ADDRESS                       ;DRB001
(1)  000504' 177777                             177777          ;TRACK/SECTOR ADDRESS                   ;DRB001
(1)  000506' 177777                             177777          ;CYLINDER ADDRESS                       ;DRB001
(1)  000510' 177777                             177777          ;TRACK/SECTOR ADDRESS                   ;DRB001
(1)  000512' 177777                             177777          ;CYLINDER ADDRESS                       ;DRB001
(1)  000514' 177777                             177777          ;TRACK/SECTOR ADDRESS                   ;DRB001
(1)  000516' 177777                             177777          ;CYLINDER ADDRESS                       ;DRB001
(1)  000520' 177777                             177777          ;TRACK/SECTOR ADDRESS                   ;DRB001
(1)  000522' 177777                             177777          ;CYLINDER ADDRESS                       ;DRB001
(1)  000524' 177777                             177777          ;TRACK/SECTOR ADDRESS                   ;DRB001
(1)  000526' 177777                             177777          ;CYLINDER ADDRESS                       ;DRB001
(1)  000530' 177777                             177777          ;TRACK/SECTOR ADDRESS                   ;DRB001
(1)  000532' 177777                             177777          ;CYLINDER ADDRESS                       ;DRB001
(1)  000534' 177777                             177777          ;TRACK/SECTOR ADDRESS                   ;DRB001
(1)  000536' 177777                             177777          ;CYLINDER ADDRESS                       ;DRB001
(1)  000540' 177777                             177777          ;TRACK/SECTOR ADDRESS                   ;DRB001
(1)  000542' 177777                             177777          ;CYLINDER ADDRESS                       ;DRB001
(1)  000544' 177777                             177777          ;TRACK/SECTOR ADDRESS                   ;DRB001
(1)  000546' 177777                             177777          ;CYLINDER ADDRESS                       ;DRB001
(1)  000550' 177777                             177777          ;TRACK/SECTOR ADDRESS                   ;DRB001
381  000552'                            ENDBBK:                                                         ;DRB001
382                                                                                                     ;DRB001
383           000004                    DLATE   = BIT2                                                  ;DRB001
384           000020                    BPORT   = BIT4                                                  ;DRB001
385           000040                    NORAND  = BIT5                                                  ;DRB001
386           000100                    BDSPT   =BIT6                                                   ;DRB001
387           000200                    RH70    = BIT7                                                  ;DRB001
388           137400                    HRDCS2  =137400
389           177777                    HRDER1  =177777
390           167777                    HRDER2  =167777
391           140153                    HRDER3  =140153
```

```
                                                                         ;**-16                ;DRB001
 000552' 000000  000000  000000  S:       0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
 000560' 000000  000000  000000
 000566' 000000  000000  000000
 000574' 000000  000000  000000
 000602' 000000  000000  000000
 000610' 000000  000000  000000
 000616' 000000  000000  000000
 000624' 000000
                                 ;DO NOT CHANGE THE ORDER OF THE NEXT 4 LOCATIONS
                                 ;NEEDED FOR MAP22 ROUTINE
 000626' 000000                  PA18:    0
 000630' 000000                  XMEM:    0
 000632' 000000                  PA22:    0
 000634' 000000                  EA22:    0
                                                                         ;**-5                 ;DRB001
 000636' 000252'                 NXTBBK: BADSPT          ; NEXT OPEN BADSPOT IN TABLE          ;DRB001
                                                                                               ;DRB001
 000640' 000000                  CYLLIM: 0               ; UPPER DISK LIMIT (DEPENDS ON DISK TYPE)      ;DRB001
 000642' 000000                  CYLNDR: 0               ;ABSOLUTE CYLINDER ADDRESS OF CURRENT OP.;DRB001
 000644' 000000                  CYL:    0               ;RELATIVE DISK ADDRESS                ;DRB001
 000646'                         DSKADR:                 ;FOR REFERENCING TRACK/SECTOR AS WORD ;DRB001
 000646'     000                 SECTOR: .BYTE   0       ;SECTOR ADDRESS OF CURRENT OPERATION  ;DRB001
 000647'     000                 TRACK:  .BYTE   0       ;TRACK ADDRESS OF CURRENT OPERATION   ;DRB001
                                                                                               ;DRB001
 000650' 000000                  BADCYL: 0               ;FROM DRIVE LOCATION REGISTERS, ON FINDING BADSPOT;DRB001
 000652' 000000                  BADSEC: 0               ;FROM RPDA ON BADSPOT OCCURANCE       ;DRB001
 000654' 000000                  BADTRK: 0               ;FROM RPDA+1 ON BADSPOT        ;DRB001
                                                                                               ;DRB001
 000656' 000000                  LASCYL: 0               ;LAST CYLINDER WHICH CAN BE WRITTEN   ;DRB001
 000660' 000000                  LASTRK: 0        ;LAST TRACK WHICH CAN BE WRITTEN             ;DRB001
 000662' 000000                  LASSEC: 0        ;LAST SECTOR WHICH CAN BE WRITTEN            ;DRB001
                                                                                               ;DRB001
 000664' 000000                  SIZTRK: 0        ;NUMBER OF TRACKS IN WBUFSZ                  ;DRB001
 000666' 000000                  SIZSEC: 0        ;REMAINDER OF WBUFSZ, IN SECTORS             ;DRB001
 000670' 000000                  FLAG:    0
 000672' 000000                  DVICE:   0
 000674' 000000                  DRIVE:   0
 000676' 000000                  UNITNO: 0
                                                                         ;**-1                 ;DRB001
 000700' 000000                  DLTCNT: 0
 000702' 000000                  FUNC:    0
 000704' 000000                  TIMER:   0
 000706' 000000                  ZERO:    0
 000710' 000000                  FERADR: 0
 000712' 000000                  CLK:     0
 000714' 000000                  TRY:     0
 000716' 000000                  STORE:   0
 000720' 000000                  HRDERR: 0
 000722' 000400                  RBUF:    .BLKW   256.

 001722'                         TABLE:
 001722' 000552'                          S
 001724' 000554'                          S+2
 001726' 000556'                          S+4
 001730' 000560'                          S+6
```



```
 001732' 000562'                          S+10
 001734' 000564'                          S+12
 001736' 000566'                          S+14
 001740' 000570'                          S+16
 001742' 000572'                          S+20
 001744' 000574'                          S+22
 001746' 000576'                          S+24
 001750' 000600'                          S+26
 001752' 000602'                          S+30
 001754' 000604'                          S+32
 001756' 000606'                          S+34
 001760' 000610'                          S+36
 001762' 000612'                          S+40
 001764' 000614'                          S+42
 001766' 000616'                          S+44
 001770' 000620'                          S+46
 001772' 000622'                          S+50
 001774' 000624'                          S+52
                                                          ;**-1                                ;DRB001
 001776' 177777                           177777
 002000' 000000                  RHCS1:   0
 002002' 000000                  RHWC:    0
 002004' 000000                  RHBA:    0
 002006' 000000                  RPDA:    0
 002010' 000000                  RHCS2:   0
 002012' 000000                  RPDS:    0
 002014' 000000                  RPER1:   0
 002016' 000000                  RPAS:    0
 002020' 000000                  RPLA:    0
 002022' 000000                  RHDB:    0
 002024' 000000                  RPMR:    0
 002026' 000000                  RPDT:    0
 002030' 000000                  RPSN:    0
 002032' 000000                  RPOF:    0
 002034' 000000                  RPDC:    0
 002036' 000000                  RPCC:    0
 002040' 000000                  RPER2:   0
 002042' 000000                  RPER3:   0
 002044' 000000                  RPEC1:   0
 002046' 000000                  RPEC2:   0
 002050' 000000                  RHBAE:   0
 002052' 000000                  RHCS3:   0
                                                          ;**-4                                ;DRB001
```

```
                        .SBTTL  MODULE CODE                                                     ;DRB001
002054' 012767  000400  176032  START:  MOV     #256.,WDTO      ;256 WORDS TO MEM/ITERATION
002062' 012767  000400  176026          MOV     #256.,WDFR      ;256 WORDS FROM MEM/ITERATION
002070' 012767  000004  176022          MOV     #4,INTR         ;4 INTERRUPTS/ITERATION
002076' 012701  000252'                 MOV     #BADSPT,R1      ;BAD SPOT TABLE                 REV D
002102' 022721  177777          4$:     CMP     #-1,(R1)+       ;END YET ?                      REV D
002106' 001375                          BNE     4$                              ;REV D
002110' 005741                          TST     -(R1)                           ;REV D
002112' 010167  176520                  MOV     R1,NXTBBK       ;RESET POINTER                  ;REV D
002116' 016767  175672  176546          MOV     DV101,DVICE     ;GET DRIVES TO TEST
002124' 123727  000041  000011          CMPB    @#41,#11        ;IF RP IS LOAD MEDIUM THEN
002132' 001021                          BNE     3$              ;BEGIN
002134' 113700  000040                  MOVB    @#40,R0         ;   GET LOAD-DEVICE NUMBER
002140' 012701  000001                  MOV     #1,R1           ;   INITIALIZE DEVICE MASK
002144' 105700                  1$:     TSTB    R0              ;   WHILE NOT POINTING AT LOAD-DEVICE DO
002146' 001403                          BEQ     2$              ;   BEGIN
002150' 006301                          ASL     R1              ;       SHIFT MASK TO NEXT DEVICE
002152' 105300                          DECB    R0              ;       KEEP TRACK OF SHIFTING
002154' 000773                          BR      1$              ;   END
002156' 130167  176510          2$:     BITB    R1,DVICE        ;   IF LOAD DEVICE IS SELECTED THEN
002162' 001405                          BEQ     3$              ;   BEGIN
002164' 113767  000040  176504          MOVB    @#40,UNITNO     ;       MOVE LOAD-DEVICE NUMBER TO DRYVE
002172' 004767  002266                  JSR     PC,DROP         ;       DROP THE LOAD-DEVICE
                                                                ;   END
002176'                         3$:                             ;END
002176' 005067  176442                  CLR     CYL             ;START AT CYLINDER 0                    ;DRB001
002202' 012767  000377  176436          MOV     #377,DSKADR     ;START AT TRACK 0, SECTOR 0             ;DRB001
                                                                ;(* INCREMENT BEFORE TEST *)             ;DRB001
002210' 004767  004500                  JSR     PC,SETUP
002214' 004767  004356                  JSR     PC,REZET        ;CLEAR THE RH
002220' 032767  000200  175570          BIT     #RH70,SR1       ;RH70 ?                                 ;OJM01
002226' 001003                          BNE     RESTRT          ;NO DONT PRI BAE & CS3                  ;OJM01
002230' 012767  177777  176364          MOV     #-1,S+50        ;SHORTEN TEMP STORAGE                   ;OJM01
002236'                         RESTRT:
                        
002236' 104415  000000' 000124'         GETPAS,BEGIN, RBUFVA             ;GET PHYSICAL ADDRESS FROM 16-BIT RBUFVA
                        
002244' 012767  177777  176424  LOOP1:  MOV     #-1,UNITNO      ;PRE-SET UNIT NUMBER
002252' 104414  000000'                 GWBUFS, BEGIN           ;GET WRITE BUFFER INFORMATION
                                                                        ;**-1                   ;DRB001
002256'                         LOOP2:
002256' 004767  000744          3$:     JSR     PC,PICKDR       ;GO PICK A DRIVE
002262' 103407                          BCS     1$              ;RETURNS HERE IF ALL DRIVES DONE        ;DRB001
002264' 004767  001572                  JSR     PC,WRTLIM       ;GET DISK LIMITS FOR THIS TYPE OF DISK  ;DRB001
002270' 005067  176420                  CLR     TRY             ;ELSE CLR RE TRY COUNT
002274' 004767  000022                  JSR     PC,CYCLE        ;GO DO A CYCLE ON THIS DRIVE
002300' 000766                          BR      LOOP2           ;DO IT TO NEXT DRIVE
                        
002302' 005767  176364          1$:     TST     DVICE           ;ANYBODY LEFT TO CHECK?
002306' 001002                          BNE     2$              ;BR IF YES
002310' 104410  000000'                 ENDS,BEGIN              ;
002314'                         2$:
(1) 002314' 104413  000000'             ENDITS,BEGIN            ;SIGNAL END OF ITERATION.
(1)                                                             ;MONITOR SHALL TEST END OF PASS
002320' 000751                          BR      LOOP1           ;BR BACK IF NO
```



```
002322'                         CYCLE:
                                                                        ;**-4                   ;DRB001
002322' 004767  000772                  JSR     PC,PICKBK       ;SELECT A SECTOR TO TEST                ;DRB001
002326' 004567  000026                  JSR     R5,WRITE        ;GO WRITE A BLOCK
002332' 004567  000116                  JSR     R5,WRITCK       ;GO DO WRITE CHECK
002336' 004567  000206                  JSR     R5,READ         ;GO READ A BLOCK
002342' 104412  000000' 000126'         CDATAS,BEGIN,RBUFPA     ; REQUEST FOR MONITOR TO CHECK DATA
(1) 002350' 002352'                     .+2                     ; IF ERROR, CONTINUE
002352' 004767  000266                  JSR     PC,RELESE       ;RELEASE THE DRIVE                      ;DRB001
002356' 000207                          RTS     PC              ;END CYCLE                              ;DRB001
```

```
                                                                        ;**-66          ;DRB001
                                        .MACRO  LINEUP  EABITS  ; LINE UP EA BITS FOR RHCS1
                                ;       LINEUP  EABITS          ; LINE UP EA BITS FOR RHCS1
                                        .NLIST
                                        MOV     EABITS,R0       ; GET EXTENDED MEMORY BITS
                                        ASL     R0              ; SHIFT 4 PLACES TO THE LEFT
                                        ASL     R0              ; TO LINE UP WITH RHCS1
                                        ASL     R0              ;
                                        ASL     R0              ;
                                        MOV     R0,XMEM         ; SAVE THE SHIFTED BITS
                                        .LIST
                                        .ENDM   LINEUP

002360' 012767  000161  176314  WRITE:  MOV     #161,FUNC       ; LOAD WRITE FUNCTION
002366' 012767  002360' 176314          MOV     #WRITE,FERADR   ;SAVE WHERE WE WERE
002374' 016746  175542                  MOV     WBUFSZ,-(SP)    ;GET WRITE SIZE
002400' 005416                          NEG     (SP)            ;NEGATE IT
002402' 012677  177374                  MOV     (SP)+,@RHWC     ; LOAD WORD COUNT
002406' 016777  175522  177370          MOV     WBUFPA,@RHBA    ; LOAD BUFFER ADDRESS
002414' 016777  176226  177364          MOV     DSKADR,@RPDA    ; LOAD DISK ADDRESS
002422' 016777  176214  177404          MOV     CYLNDR,@RPDC    ; LOAD CYLINDER ADDRESS                 ;DRB001
                                ;       LINEUP  WBUFEA          ; LINE UP EA BITS FOR RHCS1
002450' 000167  000302                  JMP     GO              ; CONTINUE
002454' 012767  000151  176220  WRITCK: MOV     #151,FUNC       ; LOAD WRITE-CHECK FUNCTION
002462' 012767  002454' 176220          MOV     #WRITCK,FERADR  ;SAVE WHERE WE WERE
002470' 016746  175446                  MOV     WBUFSZ,-(SP)    ;GET WRITE SIZE
002474' 005416                          NEG     (SP)            ;NEGATE IT
002476' 012677  177300                  MOV     (SP)+,@RHWC     ; LOAD WORD COUNT
002502' 016777  175426  177274          MOV     WBUFPA,@RHBA    ; LOAD BUFFER ADDRESS
002510' 016777  176132  177270          MOV     DSKADR,@RPDA    ; LOAD DISK ADDRESS
002516' 016777  176120  177310          MOV     CYLNDR,@RPDC    ; LOAD CYLINDER ADDRESS                 ;DRB001
                                ;       LINEUP  WBUFEA          ; LINE UP EA BITS FOR RHCS1
002544' 000167  000206                  JMP     GO              ; CONTINUE
002550' 012767  000171  176124  READ:   MOV     #171,FUNC       ; LOAD READ FUNCTION
002556' 012767  002550' 176124          MOV     #READ,FERADR    ;SAVE WHERE WE WERE
002564' 016746  175342                  MOV     RBUFSZ,-(SP)    ;GET READ SIZE
002570' 005416                          NEG     (SP)            ;NEGATE IT
002572' 012677  177204                  MOV     (SP)+,@RHWC     ; LOAD WORD COUNT
002576' 016777  175324  177200          MOV     RBUFPA,@RHBA    ; LOAD BUFFER ADDRESS
002604' 016777  176036  177174          MOV     DSKADR,@RPDA    ; LOAD DISK ADDRESS
002612' 016777  176024  177214          MOV     CYLNDR,@RPDC    ; LOAD CYLINDER ADDRESS                 ;DRB001
                                ;       LINEUP  RBUFEA          ; LINE UP EA BITS FOR RHCS1
002640' 000167  000112                  JMP     GO              ; CONTINUE
002644' 016777  176026  177136  RELESE: MOV     UNITNO,@RHCS2   ;SET UP UNIT TO RELEASE                 ;DRB001
002652' 012777  000013  177120          MOV     #13,@RHCS1      ;EXECUTE RELEASE COMMAND                ;DRB001
002660' 104407  000000'                 BREAKS,BEGIN            ;TEMPORARY RETURN TO MONITOR....
(1) 002664' 104407  000000'             BREAKS,BEGIN            ;THEN CONTINUE AT NEXT INSTRUCTION.
002670' 000207                          RTS     PC              ;IT'S RELEASED                          ;DRB001
                                                                        ;**-33          ;DRB001

002672' 016777  176000  177110  CLEAR:  MOV     UNITNO,@RHCS2   ; LOAD UNIT ADDRESS
002700' 012777  000011  177072          MOV     #11,@RHCS1      ; ISSUE A DRIVE CLEAR
002706' 000240                          NOP                     ;WAIT
002710' 000240                          NOP                     ;FOR DRIVE CLEAR TO FINISH
002712' 012777  000023  177060          MOV     #23,@RHCS1      ;ISSUE A PACK ACK
```



```
002720' 105777  177054  1$:     TSTB    @RHCS1
002724' 100401                  BMI     2$
002726' 000774                  BR      1$              ; NO, WAIT TILL DONE
002730' 017746  177062  2$:     MOV     @RPAS,-(SP)     ;CLEAR AS BIT
002734' 012677  177056          MOV     (SP)+,@RPAS
002740' 012777  040000  177032          MOV     #BIT14,@RHCS1   ; CLEAR ANY CONTROLLER ERRORS
002746' 012777  010000  177056          MOV     #BIT12,@RPOF    ; SET BIT FOR 11 FORMAT
002754' 000205                          RTS     R5              ; RETURN

002756' 016777  175714  177024  GO:     MOV     UNITNO,@RHCS2   ; LOAD UNIT SELECT
002764' 032767  001000  175064          BIT     #ADDR22,RCS1    ;22 BIT SUPPORT?                        ;DRB001
002772' 001434                          BEQ     1$              ;NO                                     ;DRB001
002774' 017767  177004  175624          MOV     @RHBA,PA18      ;GET 18 BIT ADDR
003002' 006267  175622                  ASR     XMEM            ;SHIFT EA BITS TO POSITION 4,5
003006' 006267  175616                  ASR     XMEM
003012' 006267  175612                  ASR     XMEM
003016' 006267  175606                  ASR     XMEM
003022' 104416  000000' 000626'         MAP22$, BEGIN,PA18              ; GET 22-BIT ADDR FROM 18-BIT ADDR
003030' 016777  175576  176746          MOV     PA22,@RHBA      ;LOAD BA REG
003036' 016777  175572  177004          MOV     EA22,@RHBAE     ;LOAD BAE REG
003044' 042767  000034  175562          BIC     #34,EA22        ;CLEAR UNWANTED BITS
003052' 000367  175556                  SWAB    EA22            ;LOAD INTO BITS 8,9
003056' 016767  175552  175544          MOV     EA22,XMEM       ;LOAD XMEM TO SET INTO FUNCTION CODE
003064' 056767  175540  175610  1$:     BIS     XMEM,FUNC       ; LOAD EXTENDED MEMORY BITS
003072' 016777  175604  176700          MOV     FUNC,@RHCS1     ; EXECUTE THE FUNCTION
003100' 104400  000000'                 EXITS,BEGIN             ;EXIT TO MONITOR, MODULE WAIT FOR INTERRUPT.
003104' 010046          NTRUPT: MOV     R0,-(SP)        ;SAVE R0
003106' 016700  175564          MOV     UNITNO,R0
003112' 116000  003310'         MOVB    BITTAB(R0),R0
003116' 017746  176674          MOV     @RPAS,-(SP)
003122' 040016                  BIC     R0,(SP)
003124' 012677  176666          MOV     (R6)+,@RPAS
003130' 012600                  MOV     (SP)+,R0
                                ;---------------------------------------------------------------
(1) 003132' 000004  000000' 003140'     PIRQS,BEGIN,1$          ; QUEUE UP TO CONTINUE AT 1$ AND RTI
(1)                             ;---------------------------------------------------------------

003140' 004767  002656  1$:     JSR     PC,ERRORS       ;GO CHECK FOR ERRORS
003144' 103401                  BCS     2$              ;ERRORS DETECTED                        ;DRB001
003146' 000205                  RTS     R5              ;OTHERWISE,  RETURN OK
003150'                 2$:

003150' 105767  175544          TSTB    HRDERR          ;DON'T RETRY ON HARD ERROR      REV D
003154' 100415                  BMI     4$                                              ;DRB001
003156' 005267  175532          INC     TRY             ;COUNT AN ERROR
003162' 026727  175526  000003  CMP     TRY,#3          ;TOO MANY FOR THIS CYCLE?               ;DRB001
003170' 002002                  BGE     5$              ;BR IF SO
003172' 000177  175512          JMP     @FERADR         ;RE-EXECUTE THE DRIVER ROUTINE          ;DRB001
003176' 104403  000000' 011106' 5$:     MSGWS,BEGIN,EXCED
003204' 004767  002664          JSR     PC,BAD          ;UPDATE BAD BLOCK INFO.
003210' 012605          4$:     MOV     (SP)+,R5        ;RESTORE R5
                                                                        ;**-7           ;DRB001
003212' 004767  000102  3$:     JSR     PC,PICKBK       ;TRY A DIFFERENT BLOCK
003216' 005367  175454          DEC     UNITNO          ;WANT TO RE-DO SAME DRIVE WERE ON
003222' 000167  177030          JMP     LOOP2           ;GO DO IT
                                                                        ;**-2           ;DRB001
```

```

                                 ;       ------------------------

                                                                         ;**-17                ;DRB001

003226'                          PICKDR:
003226' 005267  175444           1$:     INC     UNITNO          ;POINT TO NEXT DRIVE
003232' 026727  175440  000010           CMP     UNITNO,#8.              ;DONE LOOKING?
003240' 001002                           BNE     2$              ;BR IF NO, ELSE
003242' 000261                           SEC                     ;SET CARRY FOR NO-MORE-DRIVES         ;DRB001
003244' 000420                           BR      5$              ;EXIT ROUTINE                         ;DRB001
003246' 016700  175424           2$:     MOV     UNITNO,R0       ;USE AS AN INDEX
003252' 136067  003310' 175412           BITB    BITTAB(R0),DVICE        ;TEST THIS DEVICE?
003260' 001001                           BNE     3$              ;BR IF YES, ELSE
003262' 000761                           BR      1$
003264'                          3$:
                                                                 ;**-1                         ;DRB001
003264' 004567  003132                   JSR     R5,READY        ;SEE IF DRIVE IS READY
003270' 103402                           BCS     4$              ;BR IF IT WAS READY                   ;DRB001
003272' 004767  002730                   JSR     PC,NOTRDY       ;ELSE GO CLEAR IT AND CHECK AGAIN
003276' 004767  000700           4$:     JSR     PC,GETDVT       ;IS THIS LEGAL DRIVE TYPE?            ;DRB001
003302' 103751                           BCS     1$              ;ILLEGAL TYPE: DROPPED.  TRY NEXT     ;DRB001
003304' 000241                           CLC                     ;RETURN, NEXT DRIVE AVAILABLE         ;DRB001
003306' 000207                   5$:     RTS     PC              ;GO HOME                              ;DRB001
003310' 001001                   BITTAB: 1001
003312' 004004                           4004
003314' 020020                           20020
003316' 100100                           100100


003320'                          PICKBK:                                 ;**-11                ;DRB001
                                         ;(* FUNCTION: SELECT NEXT DISK ADDRESS TO TEST.       *)     ;DRB001
                                         ;(* AVOID WRITING OVER KNOWN BAD SPOTS OR DISK'S BAD  *)     ;DRB001
                                         ;(* SPOT FILE (IF OPERATOR TELLS US IT'S THERE).  CYL *)     ;DRB001
                                         ;(* IS "RELATIVE ADDRESS", MAX RANGE 0-410 (RP04/5) OR*)     ;DRB001
                                         ;(* 0-407 (RP06);  THIS IS MOVED TO CYLNDR AND INCRE- *)     ;DRB001
                                         ;(* MENTED TO SECOND HALF IF THIS COPY OF RPD IS IN   *)     ;DRB001
                                         ;(* PORT B MODE.                                       *)     ;DRB001
                                                                                               ;DRB001
003320' 032767  000040  174470           BIT     #NORAND,SR1     ;IF SEQUENTIAL SEEKS THEN             ;DRB001
003326' 001403                           BEQ     1$              ;BEGIN                                ;DRB001
003330' 105267  175312                   INCB    SECTOR          ; SECTOR := SECTOR + 1                ;DRB001
003334' 000432                           BR      2$              ;END                                  ;DRB001
003336'                          1$:                             ;ELSE (* RANDOM SEEKS *)              ;DRB001
                                                                 ;BEGIN (* FUDGE 3 UN-RELATED          ;DRB001
                                                                 ;  (* NUMBERS FROM 1 RAND CALL  *)    ;DRB001
003336' 104417  000000'                  RANDS,BEGIN
003342' 016700  174506                   MOV     RANNUM,R0       ; R0 := RANNUM                        ;DRB001
003346' 042700  176000                   BIC     #176000,R0      ; R0 := R0 MOD 1024.                  ;DRB001
003352' 010067  175266                   MOV     R0,CYL          ; CYL := R0                           ;DRB001
003356' 016700  174472                   MOV     RANNUM,R0       ; "RECHARGE" R0                       ;DRB001
003362' 000300                           SWAB    R0              ; R0<0:8> := R0<8:8>                  ;DRB001
003364' 042700  177740                   BIC     #177740,R0      ; R0 := R0 MOD 32.                    ;DRB001
003370' 110067  175252                   MOVB    R0,SECTOR       ; SECTOR := R0                        ;DRB001
003374' 016700  174454                   MOV     RANNUM,R0       ; AND AGAIN                           ;DRB001
003400' 000241                           CLC                     ; CARRY := 0                          ;DRB001
```



```
003402' 006100                           ROL     R0              ; CARRY := R0<15:1>                   ;DRB001
003404' 006100                           ROL     R0                                                    ;DRB001
003406' 006100                           ROL     R0              ;                                     ;DRB001
003410' 006100                           ROL     R0              ;                                     ;DRB001
003412' 042700  177740                   BIC     #177740,R0      ; R0 := (R0<0:13>*8.) + R0<13:3>      ;DRB001
                                                                 ; (* NOTE: THE FORMULAE USED ARE *)   ;DRB001
                                                                 ; (* ABSOLUTELY IRRELEVANT       *)   ;DRB001
003416' 110067  175225                   MOVB    R0,TRACK        ; TRACK := R0                         ;DRB001
003422'                          2$:                             ;END                                  ;DRB001
003422' 126727  175220  000025           CMPB    SECTOR,#21.     ;IF SECTOR GT 21. THEN                ;DRB001
003430' 003412                           BLE     3$              ;BEGIN                                ;DRB001
003432' 105267  175211                   INCB    TRACK           ; TRACK := TRACK + 1                  ;DRB001
003436' 016700  175204                   MOV     SECTOR,R0       ; R0 := SECTOR                        ;DRB001
003442' 012701  000026                   MOV     #22.,R1         ; R1 := # OF SECTORS IN TRACK         ;DRB001
003446' 004767  000760                   JSR     PC,MODLUS       ; R0 := R0 MOD R1                     ;DRB001
003452' 110067  175170                   MOVB    R0,SECTOR       ; SECTOR := R0                        ;DRB001
003456'                          3$:                             ;END                                  ;DRB001
                                         ;(* NOW, CUT VALUES DOWN TO LEGAL RANGES FOR CURRENT DRIVE TYPE *) ;DRB001
003456' 126727  175165  000022           CMPB    TRACK,#18.      ;IF TRACK GT 18. THEN                 ;DRB001
003464' 003412                           BLE     4$              ;BEGIN                                ;DRB001
003466' 005267  175152                   INC     CYL             ; CYL := CYL + 1                      ;DRB001
003472' 116700  175151                   MOVB    TRACK,R0        ; R0 := TRACK                         ;DRB001
003476' 012701  000023                   MOV     #19.,R1         ; R1 := 19.                           ;DRB001
003502' 004767  000724                   JSR     PC,MODLUS       ; R0 := R0 MOD R1                     ;DRB001
003506' 110067  175135                   MOVB    R0,TRACK        ; TRACK := R0                         ;DRB001
003512'                          4$:                             ;END                                  ;DRB001
003512' 026767  175126  175120           CMP     CYL,CYLLIM      ;IF CYL GT # OF CYLINDERS IN RANGE THEN ;DRB001
003520' 002420                           BLT     5$              ;BEGIN                                ;DRB001
003522' 032767  000040  174266           BIT     #NORAND,SR1     ; IF SEQUENTIAL SEEKS THEN            ;DRB001
003530' 001404                           BEQ     6$              ; BEGIN                               ;DRB001
003532' 005067  175106                   CLR     CYL             ;  RESET CYLINDER                     ;DRB001
003536' 005067  175104                   CLR     DSKADR          ;  TRACK := 0, SECTOR := 0            ;DRB001
003542'                          6$:                             ; ELSE                                ;DRB001
                                                                 ; BEGIN                               ;DRB001
003542' 016700  175076                   MOV     CYL,R0          ;  R0 := CYL                          ;DRB001
003546' 016701  175066                   MOV     CYLLIM,R1       ;  R1 := CYLLIM                       ;DRB001
003552' 004767  000654                   JSR     PC,MODLUS       ;  R0 := R0 MOD R1                    ;DRB001
003556' 010067  175062                   MOV     R0,CYL          ;  CYL := R0                          ;DRB001
003562'                          7$:                             ; END                                 ;DRB001
003562'                          5$:                             ;END                                  ;DRB001
003562' 016767  175056  175052           MOV     CYL,CYLNDR      ;CYLNDR := CYL                        ;DRB001
003570' 032767  000020  174220           BIT     #BPORT,SR1      ;IF BPORT AND DUAL PORT THEN
003576' 001407                           BEQ     8$              ;BEGIN                                ;DRB001
003600' 032767  000001  175062           BIT     #BIT0,FLAG      ;                                     ;DRB001
003606' 001403                           BEQ     8$              ;                                     ;DRB001
003610' 066767  175024  175024           ADD     CYLLIM,CYLNDR   ; INCREMENT TO SECOND HALF OF DISK    ;DRB001
003616'                          8$:                             ;END                                  ;DRB001
003616' 026767  175020  175032           CMP     CYLNDR,LASCYL   ;IF CYLNDR GT LASCYL THEN             ;DRB001
003624' 003431                           BLE     20$             ;BEGIN          ;REV. D0
003626' 016767  175024  175006           MOV     LASCYL,CYLNDR   ; CYLNDR := LASCYL (* TRUNCATE *)     ;DRB001
003634' 032767  000040  174154           BIT     #NORAND,SR1     ; IF SEQUENTIAL THEN                  ;DRB001
003642' 001404                           BEQ     10$             ; BEGIN                               ;DRB001
003644' 005067  174774                   CLR     CYL             ;  RESET CYLINDER                     ;DRB001
003650' 005067  174772                   CLR     DSKADR          ;  TRACK := 0, SECTOR := 0            ;DRB001
003654'                          10$:                            ; END                                 ;DRB001
003654' 126767  174767  174776           CMPB    TRACK,LASTRK    ; IF TRACK GT LASTRKK THEN            ;DRB001
```

```
003662' 003412                  BLE     11$             ; BEGIN                                 ;DRB001
003664' 116767  174770  174755  MOVB    LASTRK,TRACK    ;  TRACK := LASTRK                      ;DRB001
003672' 126767  174750  174762  CMPB    SECTOR,LASSEC   ;  IF SECTOR GT LASSEC THEN             ;DRB001
003700' 003403                  BLE     12$             ;  BEGIN                                ;DRB001
003702' 116767  174754  174736  MOVB    LASSEC,SECTOR   ;   SECTOR := LASSEC                    ;DRB001
003710'                         12$:                    ;  END                                  ;DRB001
003710'                         11$:                    ; END                                   ;DRB001

003710'                         20$:                                                            ;DRB001
                                                                                                ;DRB001
                                ;(* NOW, CHECK BADSPOT TABLE *)                                 ;DRB001
003710' 012700  000252'         MOV     #BADSPT,R0      ;R0 := ADDRESS OF TABLE                 ;DRB001
003714' 016701  174716          MOV     NXTBBK,R1       ;R1 := END OF TABLE                     ;DRB001
003720' 116702  174722          MOVB    SECTOR,R2       ;R2 := SECTOR                           ;DRB001
003724' 116703  174717          MOVB    TRACK,R3        ;R3 := TRACK                            ;DRB001
003730' 016704  174706          MOV     CYLNDR,R4       ;R4 := CYLNDR                           ;DRB001
                                ;(* CALCULATE UPPER LIMIT OF THIS WRITE *)                     ;DRB001
003734' 066702  174726          ADD     SIZSEC,R2       ;R2 := R2 + # OF SECTORS IN WRITE       ;DRB001
003740' 020227  000025          CMP     R2,#21.         ;IF R2 GT 21. THEN                      ;DRB001
003744' 003403                  BLE     13$             ;BEGIN                                  ;DRB001
003746' 005203                  INC     R3              ; R3 := R3 + 1 (INCR. UPPER TRACK)      ;DRB001
003750' 162702  000026          SUB     #22.,R2         ; TRUNCATE R2                           ;DRB001
003754'                         13$:                    ;END                                    ;DRB001
003754' 066703  174704          ADD     SIZTRK,R3       ;R3 := R3 + TRKSIZ                      ;DRB001
003760' 020327  000022          CMP     R3,#18.         ;IF R3 GT 18. THEN                      ;DRB001
003764' 003403                  BLE     14$             ;BEGIN                                  ;DRB001
003766' 005204                  INC     R4              ; R4 := R4 + 1 (INCR. UPPER CYLINDER)   ;DRB001
003770' 162703  000023          SUB     #19.,R3         ; TRUNCATE R3                           ;DRB001
003774'                         14$:                    ;END                                    ;DRB001
003774' 020001                  15$:    CMP     R0,R1   ;WHILE R0 LT R1 DO                      ;DRB001
003776' 002030                  BGE     16$             ;BEGIN                                  ;DRB001
004000' 026710  174636          CMP     CYLNDR,(R0)     ; IF CYLNDR LE (R0) AND R4 GE (R0)      ;DRB001
004004' 003022                  BGT     17$             ;  AND IRACK LE 3(R0) AND R3 GE 3(R0)   ;DRB001
004006' 020410                  CMP     R4,(R0)         ;  AND SECTOR LE 2(R0) AND R2 GE 2(R0) THEN;DRB001
004010' 002420                  BLT     17$             ;                                       ;DRB001
004012' 126760  174631  000003  CMPB    TRACK,3(R0)     ;                                       ;DRB001
004020' 003014                  BGT     17$             ;                                       ;DRB001
004022' 120360  000003          CMPB    R3,3(R0)        ;                                       ;DRB001
004026' 002411                  BLT     17$             ;                                       ;DRB001
004030' 126760  174612  000002  CMPB    SECTOR,2(R0)    ;                                       ;DRB001
004036' 003005                  BGT     17$             ;                                       ;DRB001
004040' 120260  000002          CMPB    R2,2(R0)        ;                                       ;DRB001
004044' 002402                  BLT     17$             ; BEGIN                                 ;DRB001
004046' 000167  177246          JMP     PICKBK          ;  TRY ANOTHER SECTOR                   ;DRB001
004052'                         17$:                    ; END                                   ;DRB001
004052' 062700  000004          ADD     #4,R0           ; INCREMENT TO NEXT BAD SPOT            ;DRB001
004056' 000746                  BR      15$             ;END                                    ;DRB001
004060'                         16$:                    ;                                       ;DRB001
004060' 000207                  RTS     PC              ;CPU GO HOME!!!!                        ;DRB001
                                                                                                ;DRB001
                                                                                                ;DRB001
004062'                         WRTLIM:                                                         ;DRB001
                                ;(* WE ALLOW USER TO CHANGE WBUFSZ "AT WILL"--THEREFORE *)      ;DRB001
                                ;(* WE DO NOT AUTOMATICALLY KNOW THE TRANSFER SIZE.     *)      ;DRB001
                                ;(* COMPUTATION IS MADE EASIER, FURTHERMORE, IF WE HAVE *)      ;DRB001
                                ;(* THE SIZE BROKEN INTO TRACKS AND SECTORS (SINCE      *)      ;DRB001
```



```
                                ;(* WBUFSZ IS ONE WORD, THE TRANSFER CAN'T BE GREATER   *)      ;DRB001
                                ;(* THAN 12 TRACKS).  THIS ROUTINE ALSO CALCULATES THE  *)      ;DRB001
                                ;(* UPPER LIMITS ON DISK WRITES SO AS NOT TO OVERFLOW    *)      ;DRB001
                                ;(* HIGH CYLINDER.                                      *)      ;DRB001
                                                                                                ;DRB001
004062' 016700  174054          MOV     WBUFSZ,R0       ;R0 := ALLOCATED WRITE BUFFER           ;DRB001
004066' 000300                  SWAB    R0              ;R0 := R0 / 256.                        ;DRB001
004070' 042700  177400          BIC     #177400,R0      ; (* R0 IS WBUFSZ IN SECTORS *)         ;DRB001
004074' 005067  174564          CLR     SIZTRK          ;SIZTRK := 0                            ;DRB001
004100' 020027  000026          1$:     CMP     R0,#22. ;WHILE R0 GE 22. DO                     ;DRB001
004104' 002405                  BLT     2$              ;BEGIN                                  ;DRB001
004106' 105267  174552          INCB    SIZTRK          ; SIZTRK := SIZTRK + 1                  ;DRB001
004112' 162700  000026          SUB     #22.,R0         ; DECREMENT SECTORS BY 1 TRACK          ;DRB001
004116' 000770                  BR      1$              ;END                                    ;DRB001
004120' 010067  174542          2$:     MOV     R0,SIZSEC       ;SIZSEC := REMAINDER            ;DRB001
004124' 012701  000022          MOV     #18.,R1         ;R1 := HIGH TRACK                       ;DRB001
004130' 012702  000025          MOV     #21.,R2         ;R2 := HIGH SECTOR                      ;DRB001
004134' 160002                  SUB     R0,R2           ;R2 := R2 - SIZSEC                      ;DRB001
004136' 005702                  TST     R2              ;IF R2 LT 0 THEN                        ;DRB001
004140' 002003                  BGE     3$              ;BEGIN                                  ;DRB001
004142' 062702  000026          ADD     #22.,R2         ; INCREMENT BACK UP                     ;DRB001
004146' 005301                  DEC     R1              ; CAN'T FIT THAT LAST TRACK...          ;DRB001
004150'                         3$:                     ;END                                    ;DRB001
004150' 166701  174510          SUB     SIZTRK,R1       ;R1 := R1 - SIZTRK                      ;DRB001
004154' 005701                  TST     R1              ;IF R1 LT 0 THEN                        ;DRB001
004156' 002004                  BGE     4$              ;BEGIN                                  ;DRB001
004160' 062701  000023          ADD     #19.,R1         ; INCREMENT R1 BACK UP                  ;DRB001
004164' 005367  174466          DEC     LASCYL          ; LASCYL := LASCYL - 1                  ;DRB001
004170'                         4$:                     ;END                                    ;DRB001
004170' 010167  174464          MOV     R1,LASTRK       ;LASTRK := HIGHEST TRACK WE CAN WRITE   ;DRB001
004174' 010267  174462          MOV     R2,LASSEC       ;LASSEC := HIGHEST SECTOR WE CAN WRITE  ;DRB001
004200' 000207                  RTS     PC              ;END OF ROUTINE                         ;DRB001
004202'                         GETDVT:                                                         ;DRB001
                                ;(* FIND WHICH TYPE OF DRIVE THIS IS.  IF RP04/5, SET CYLLIM *) ;DRB001
                                ;(* TO 206.  IF RP06, SET CYLLIM TO 408.  DROP THE DRIVE IF  *) ;DRB001
                                ;(* IT IS NOT A RP04/5/6                                   *) ;DRB001
                                                                                                ;DRB001
004202' 016777  174470  175600  MOV     UNITNO,@RHCS2   ;SELECT THE DRIVE                       ;DRB001
004210' 027727  175612  024022  CMP     @RPDT,#24022    ;IF IT IS AN RP06 DUAL PORT THEN        ;DRB001
004216' 001012                  BNE     1$              ;BEGIN                                  ;DRB001
004220' 052767  000001  174442  BIS     #BIT0,FLAG      ; DUAL PORT                             ;DRB001
004226' 012767  000627  174404  MOV     #407.,CYLLIM    ; CYLLIM := 408.                        ;DRB001
004234' 012767  001455  174414  MOV     #813.,LASCYL    ; LASCYL := DISK HIGH LIM               ;DRB001
004242' 000471                  BR      2$              ;END                                    ;DRB001
004244'                         1$:                     ;ELSE                                   ;DRB001
004244' 027727  175556  024021  CMP     @RPDT,#24021    ;IF IT IS RP04/5 DUAL PORT THEN         ;DRB001
004252' 001404                  BEQ     3$              ;                                       ;DRB001
004254' 027727  175546  024020  CMP     @RPDT,#24020    ;                                       ;DRB001
004262' 001012                  BNE     4$              ;BEGIN                                  ;DRB001
004264'                         3$:                                                             ;DRB001
004264' 052767  000001  174376  BIS     #BIT0,FLAG      ; DUAL PORT                             ;DRB001
004272' 012767  000315  174340  MOV     #205.,CYLLIM    ; CYLLIM := 206.                        ;DRB001
004300' 012767  000631  174350  MOV     #409.,LASCYL    ; LASCYL := DISK HIGH LIM               ;DRB001
004306' 000447                  BR      5$              ;END                                    ;DRB001
004310'                         4$:                     ;ELSE                                   ;DRB001
004310' 027727  175512  020022  CMP     @RPDT,#20022    ;IF RP06 SINGLE PORT THEN               ;DRB001
```

```
    004316' 001012                          BNE     6$              ;BEGIN                          ;DRB001
    004320' 042767  000001  174342          BIC     #BIT0,FLAG      ; SINGLE PORT                   ;DRB001
    004326' 012767  001456  174304          MOV     #814.,CYLLIM    ; CAN USE ENTIRE DISK           ;DRB001
    004334' 012767  001455  174314          MOV     #813.,LASCYL    ; LASCYL := DISK HIGH LIMIT     ;DRB001
    004342' 000431                          BR      2$              ;END                            ;DRB001
    004344'                         6$:                             ;ELSE                           ;DRB001
    004344' 027727  175456  020021          CMP     @RPDT,#20021    ;IF RP04/5 SINGLE PORT THEN     ;DRB001
    004352' 001404                          BEQ     8$              ;                               ;DRB001
    004354' 027727  175446  020020          CMP     @RPDT,#20020    ;                               ;DRB001
    004362' 001012                          BNE     9$              ;                               ;DRB001
    004364'                         8$:                             ;BEGIN                          ;DRB001
    004364' 042767  000001  174276          BIC     #BIT0,FLAG      ; SINGLE PORT                   ;DRB001
    004372' 012767  000632  174240          MOV     #410.,CYLLIM    ; USE ENTIRE DISK               ;DRB001
    004400' 012767  000631  174250          MOV     #409.,LASCYL    ; LASCYL := DISK HIGH LIMIT     ;DRB001
    004406' 000407                          BR      2$              ;END                            ;DRB001
    004410'                         9$:                             ;ELSE                           ;DRB001
                                                                    ;BEGIN                          ;DRB001
    004410' 004767  000050                  JSR     PC,DROP         ; DROP THE DRIVE                ;DRB001
    004414' 104403  000000' 011176'         MSGN$,BEGIN,BADTYP      ;ASCII MESSAGE CALL WITH COMMON HEADER
    004422' 000261                          SEC                     ; ERROR RETURN                  ;DRB001
    004424' 000401                          BR      7$              ;                               ;DRB001
    004426'                         5$:                             ;END                            ;DRB001
    004426'                         2$:                             ;                               ;DRB001
    004426' 000241                          CLC                     ;NO ERROR                       ;DRB001
    004430'                         7$:                             ;                               ;DRB001
    004430' 000207                          RTS     PC              ;RETURN                         ;DRB001
    004432'                         MODLUS:                                                         ;DRB001
                                            ;(* THIS LITTLE ROUTINE COMPUTES R0 MODULUS R1      *)  ;DRB001
                                            ;(*  I.E., THE REMAINDER OF R0/R1, AND RETURNS THE  *)  ;DRB001
                                            ;(* ANSWER IN R0.                                   *)  ;DRB001
    004432' 020001                  1$:     CMP     R0,R1           ;WHILE R0 GE R1 DO              ;DRB001
    004434' 002402                          BLT     2$              ;BEGIN                          ;DRB001
    004436' 160100                          SUB     R1,R0           ; R0 := R0-R1                   ;DRB001
    004440' 000774                          BR      1$              ;END                            ;DRB001
    004442'                         2$:                             ;                               ;DRB001
    004442' 000207                          RTS     PC              ;THAT'S ALL                     ;DRB001

    004444' 012700  000722'         CLRRB:  MOV     #RBUF,R0        ;CLEAR RBUF BUFFER
    004450' 016701  173456                  MOV     RBUFSZ,R1       ;GET ITS ADDR AND SIZE
    004454' 005020                  CLRCOM: CLR     (R0)+           ;CLEAR ANOTHER
    004456' 005301                          DEC     R1              ;COUNT ANOTHER
    004460' 001375                          BNE     CLRCOM          ;BR BACK TILL DONE
    004462' 000207                          RTS     PC

    004464' 012701  000001          DROP:   MOV     #1,R1           ; INITIALIZE DROP PICKER
    004470' 016700  174202                  MOV     UNITNO,R0       ; GET THE DRIVE NUMBER
    004474' 001403                          BEQ     2$              ; IF DRIVE 0 GO DROP IT
    004476' 006301                  1$:     ASL     R1              ; POINT TO NEXT DRIVE
    004500' 005300                          DEC     R0              ; IS THIS THE ONE ?
    004502' 001375                          BNE     1$              ; NO, LOOK AGAIN
    004504' 040167  174162          2$:     BIC     R1,DVICE        ; DROP THE DRIVE
                                            ;**************************************************************
(1)                                                                 ;CONVERT UNITNO TO ASCII AND
(1)                                                                 ;STORE AT ADR1
(1) 004510' 104420  000000' 000676'         OTOA$,BEGIN,UNITNO,ADR1
(1) 004516' 011374'
```



```
                                            ;**************************************************************
(1)
    004520' 000207                          RTS     PC              ; RETURN
                                    ;       -------------------------






                                            ;////\\\\////\\\\////\\\\////\\\\////\\\\////\\\\////\\\\////\\\\
                                            ;////\\\\////\\\\////\\\\////\\\\////\\\\////\\\\////\\\\////\\\\

                                                    ;R E V  D       ;MCP






    004522' 005067  000426          EROR:   CLR     DLTFLG          ;INITIALIZE DLT FLAG
    004526' 104403  000000' 011360'         MSGN$,BEGIN,ASTER       ;ASCII MESSAGE CALL WITH COMMON HEADER
    004534' 005067  174160                  CLR     HRDERR                  ;RESET ERROR INDICATOR
    004540' 032767  100000  174014          BIT     #BIT15,S+10             ;
    004546' 001417                          BEQ     14$                     ;
    004550' 005267  174124                  INC     DLTCNT                  ;
    004554' 052767  000001  000372          BIS     #1,DLTFLG               ;SET FLAG FOR DATA LATE
    004562' 032767  000004  173226          BIT     #DLATE,SR1              ;
    004570' 001006                          BNE     14$                     ;
    004572' 052767  177777  174120          BIS     #-1,HRDERR              ;
    004600' 104403  000000' 011112'         MSGN$,BEGIN,DLTERR      ;ASCII MESSAGE CALL WITH COMMON HEADER
    004606' 032777  020000  175164  14$:    BIT     #BIT13,@RHCS1           ;MASSBUS CONTROL PARITY ERR?
    004614' 001406                          BEQ     5$
    004616' 052767  177777  174074          BIS     #-1,HRDERR              ;SET HARD ERR
    004624' 104403  000000' 011064'         MSGN$,BEGIN,MCPERR      ;ASCII MESSAGE CALL WITH COMMON HEADER
    004632' 032767  137400  173722  5$:     BIT     #HRDCS2,S+10            ;ANY ERRORS ?
    004640' 001402                          BEQ     1$
    004642' 004767  000310                  JSR     PC,CHKCS2               ;GO SEE THE BITS
    004646' 032767  040000  173710  1$:     BIT     #BIT14,S+12             ;ANY SET IN THE ERROR REG'S ?
    004654' 001432                          BEQ     4$
    004656' 032767  177777  173702          BIT     #HRDER1,S+14
    004664' 001402                          BEQ     2$
    004666' 004767  000426                  JSR     PC,CHKER1
    004672' 032767  167777  173712  2$:     BIT     #HRDER2,S+40            ;ANY ERRORS IN ER2
    004700' 001406                          BEQ     3$
    004702' 052767  177777  174010          BIS     #-1,HRDERR
    004710' 104403  000000' 011210'         MSGN$,BEGIN,ER2ERR      ;ASCII MESSAGE CALL WITH COMMON HEADER
    004716' 032767  140153  173670  3$:     BIT     #HRDER3,S+42            ;TEST ER3
    004724' 001406                          BEQ     4$
    004726' 052767  177777  173764          BIS     #-1,HRDERR              ;SET HARD ERRROR INDICATOR
    004734' 104403  000000' 011214'         MSGN$,BEGIN,ER3ERR      ;ASCII MESSAGE CALL WITH COMMON HEADER
    004742' 032767  040000  173612  4$:     BIT     #BIT14,S+10             ; WCE ?
    004750' 001406                          BEQ     6$
    004752' 052767  000002  000174          BIS     #2,DLTFLG               ;SET FLAG FOR SOFT TO BE USED IF DLT ALSO OCCURE
    004760' 104403  000000' 011220'         MSGN$,BEGIN,WCEERR      ;ASCII MESSAGE CALL WITH COMMON HEADER
    004766' 032767  040000  173556  6$:     BIT     #BIT14,S                ;TRE  ?
    004774' 001406                          BEQ     7$
    004776' 052767  000002  000150          BIS     #2,DLTFLG               ;SET SOFT FLAG
    005004' 104403  000000' 011056'         MSGN$,BEGIN,TRERR       ;ASCII MESSAGE CALL WITH COMMON HEADER
    005012' 032767  100000  173532  7$:     BIT     #BIT15,S                ;SC ?
```

```
 993  005020' 001406                          BEQ     8$
 994  005022' 052767  000002  000124          BIS     #2,DLTFLG               ;SET FLAG
 995  005030' 104403  000000' 011224'         MSGN$,BEGIN,SCERR       ;ASCII MESSAGE CALL WITH COMMON HEADER
 996  005036' 005767  173520          8$:     TST     S+10
 997  005042' 100405                          BMI     10$
 998  005044' 016767  174752  173644          MOV     RHDB,STORE
 999  005052' 005067  174744                  CLR     RHDB
1000  005056' 005767  173636          10$:    TST     HRDERR                  ;ANNY HARD ERROR'S ?
1001  005062' 100414                          BMI     12$                     ;IF TRUE THEN BR
1002  005064' 032767  000001  000062          BIT     #1,DLTFLG               ;DLT ?  #DLATE IN SR1 MUST BE SET IF TRUE
1003  005072' 001404                          BEQ     22$
1004  005074' 022767  000003  000052          CMP     #3,DLTFLG               ;I; (SOFT & DLT NOT COUNTED )  ?
1005  005102' 001007                          BNE     11$
1006  005104'                         22$:
1007                                          ;*****************************************************************
 (1)  005104' 104406  000000' 001722'         SOFER$,BEGIN,TABLE      ;
 (1)                                          ;*****************************************************************
1008  005112' 000403                          BR      11$                     ;
1009  005114'                         12$:
 (1)                                          ;*****************************************************************
 (1)  005114' 104405  000000' 001722'         HRDER$,BEGIN,TABLE      ;
 (1)                                          ;*****************************************************************
1010  005122' 005767  174674          11$:    TST     RHDB
1011  005126' 001003                          BNE     9$
1012  005130' 016767  173562  174664          MOV     STORE,RHDB
1013  005136' 004767  001064          9$:     JSR     PC,NOTRDY
1014  005142' 000261                          SEC
1015
1016  005144' 104403  000000' 011360'         MSGN$,BEGIN,ASTER       ;ASCII MESSAGE CALL WITH COMMON HEADER
1017  005152' 000207                          RTS     PC
1018
1019  005154' 000000                  DLTFLG: .WORD   0                       ;DLT FLAG TO BE USED FOR SOFT ERROR
1020                                                                          ; & INHIBIT DLT COMPARISONS
1021
1022  005156' 052767  177777  173534  CHKCS2: BIS     #-1,HRDERR              ;SET HARD ERROR
1023  005164' 104403  000000' 011230'         MSGN$,BEGIN,CS2HD       ;ASCII MESSAGE CALL WITH COMMON HEADER
1024  005172' 032767  020000  173362  1$:     BIT     #BIT13,S+10             ;DPE ?
1025  005200' 001403                          BEQ     2$
1026  005202' 104403  000000' 011234'         MSGN$,BEGIN,DPER        ;ASCII MESSAGE CALL WITH COMMON HEADER
1027  005210' 032767  010000  173344  2$:     BIT     #BIT12,S+10             ;NED ?
1028  005216' 001403                          BEQ     3$
1029  005220' 104403  000000' 011240'         MSGN$,BEGIN,NEDER       ;ASCII MESSAGE CALL WITH COMMON HEADER
1030  005226' 032767  004000  173326  3$:     BIT     #BIT11,S+10             ;NXM ?
1031  005234' 001403                          BEQ     4$
1032  005236' 104403  000000' 011244'         MSGN$,BEGIN,NXM ;ASCII MESSAGE CALL WITH COMMON HEADER
1033  005244' 032767  002000  173310  4$:     BIT     #BIT10,S+10
1034  005252' 001403                          BEQ     5$
1035  005254' 104403  000000' 011250'         MSGN$,BEGIN,PGERR       ;ASCII MESSAGE CALL WITH COMMON HEADER
1036  005262' 032767  001000  173272  5$:     BIT     #BIT9,S+10              ;MXF ?
1037  005270' 001403                          BEQ     6$
1038  005272' 104403  000000' 011254'         MSGN$,BEGIN,MXFER       ;ASCII MESSAGE CALL WITH COMMON HEADER
1039
1040  005300' 032767  000400  173254  6$:     BIT     #BIT8,S+10
1041  005306' 001403                          BEQ     7$
1042  005310' 104403  000000' 011260'         MSGN$,BEGIN,MDPER       ;ASCII MESSAGE CALL WITH COMMON HEADER
1043  005316' 000207                  7$:     RTS     PC
```



```
1044
1045  005320'                         CHKER1:
 (1)  005320' 104403  000000' 011264'         MSGN$,BEGIN,ER1HD       ;ASCII MESSAGE CALL WITH COMMON HEADER
1046  005326' 032767  100000  173232          BIT     #BIT15,S+14             ;DATA CHECK ERROR ?
1047  005334' 001406                          BEQ     1$
1048  005336' 052767  000002  177610          BIS     #2,DLTFLG               ;SET SOFT ERR
1049  005344' 104403  000000' 011270'         MSGN$,BEGIN,DCKER       ;ASCII MESSAGE CALL WITH COMMON HEADER
1050  005352' 032767  000020  173206  1$:     BIT     #BIT4,S+14              ;FORMAT ERROR ?
1051  005360' 001423                          BEQ     2$
1052  005362' 104403  000000' 011274'         MSGN$,BEGIN,FERER       ;ASCII MESSAGE CALL WITH COMMON HEADER
1053  005370' 016767  173172  173320          MOV     S+14,STORE      ;SAVE ER1
1054  005376' 042767  077757  173312          BIC     #77757,STORE
1055  005404' 022767  100020  173304          CMP     #BIT15 ! BIT4,STORE
1056  005412' 001003                          BNE     21$
1057  005414' 052767  177777  173276          BIS     #-1,HRDERR              ;SET HARD ERROR INDICATOR
1058  005422' 052767  000002  177524  21$:    BIS     #2,DLTFLG               ;SET SOFT ERR
1059
1060  005430' 032767  000400  173130  2$:     BIT     #BIT8,S+14              ;HCRC ?
1061  005436' 001406                          BEQ     3$
1062  005440' 052767  000002  177506          BIS     #2,DLTFLG               ;SET SOFT ERR
1063  005446' 104403  000000' 011300'         MSGN$,BEGIN,HCRC        ;ASCII MESSAGE CALL WITH COMMON HEADER
1064  005454' 032767  000200  173104  3$:     BIT     #BIT7,S+14              ;HEADER COMP ERR ?
1065  005462' 001420                          BEQ     5$
1066  005464' 104403  000000' 011304'         MSGN$,BEGIN,HCE ;ASCII MESSAGE CALL WITH COMMON HEADER
1067  005472' 016767  173070  173216          MOV     S+14,STORE
1068  005500' 042767  177177  173210          BIC     #177177,STORE           ;SAVE ONLY HCRC & HCE
1069  005506' 022767  000600  173202          CMP     #600,STORE
1070  005514' 001403                          BEQ     5$                      ;SOFT ERROR IF BOTH SET
1071  005516' 052767  177777  173174          BIS     #-1,HRDERR              ;SET HRD ERR .
1072  005524' 032767  077157  173034  5$:     BIT     #77157,S+14             ;ONLY HARD'S LEFT
1073  005532' 001001                          BNE     6$
1074  005534' 000207                          RTS     PC
1075
1076  005536' 052767  177777  173154  6$:     BIS     #-1,HRDERR              ;SET HRD ERR .
1077  005544' 032767  040000  173014          BIT     #BIT14,S+14             ;UNS ?
1078  005552' 001403                          BEQ     7$
1079  005554' 104403  000000' 011310'         MSGN$,BEGIN,UNSER       ;ASCII MESSAGE CALL WITH COMMON HEADER
1080  005562' 032767  020000  172776  7$:     BIT     #BIT13,S+14             ;DPI ?
1081  005570' 001403                          BEQ     8$
1082  005572' 104403  000000' 011314'         MSGN$,BEGIN,DPIER       ;ASCII MESSAGE CALL WITH COMMON HEADER
1083  005600' 032767  010000  172760  8$:     BIT     #BIT12,S+14             ;DTE ?
1084  005606' 001403                          BEQ     9$
1085  005610' 104403  000000' 011320'         MSGN$,BEGIN,DTER        ;ASCII MESSAGE CALL WITH COMMON HEADER
1086  005616' 032767  004000  172742  9$:     BIT     #BIT11,S+14             ;WLE ?
1087  005624' 001403                          BEQ     10$
1088  005626' 104403  000000' 011324'         MSGN$,BEGIN,WLER        ;ASCII MESSAGE CALL WITH COMMON HEADER
1089  005634' 032767  002000  172724  10$:    BIT     #BIT10,S+14             ;IAE ?
1090  005642' 001403                          BEQ     20$
1091  005644' 104403  000000' 011330'         MSGN$,BEGIN,IAER        ;ASCII MESSAGE CALL WITH COMMON HEADER
1092  005652' 032767  001000  172706  20$:    BIT     #BIT9,S+14              ;               >
1093  005660' 001403                          BEQ     17$                     ;               >
1094  005662' 104403  000000' 011364'         MSGN$,BEGIN,AOE ;ASCII MESSAGE CALL WITH COMMON HEADER
1095  005670' 032767  000040  172670  17$:    BIT     #BIT5,S+14              ;
1096  005676' 001403                          BEQ     11$                     ;
1097  005700' 104403  000000' 011370'         MSGN$,BEGIN,WCF ;ASCII MESSAGE CALL WITH COMMON HEADER
1098  005706' 032767  000100  172652  11$:    BIT     #BIT6,S+14              ;ECH ?
```

```
 1099  005714' 001405                          BEQ     12$
 1100  005716' 104403  000000' 011334'         MSGN$,BEGIN,ECHR        ;ASCII MESSAGE CALL WITH COMMON HEADER
 1101  005724' 004767  000144                  JSR     PC,BAD
 1102  005730' 032767  000010  172630  12$:    BIT     #BIT3,S+14              ;PAR  ?
 1103  005736' 001403                          BEQ     13$
 1104  005740' 104403  000000' 011340'         MSGN$,BEGIN,PARR        ;ASCII MESSAGE CALL WITH COMMON HEADER
 1105  005746' 032767  000004  172612  13$:    BIT     #BIT2,S+14              ;RMR ?
 1106  005754' 001403                          BEQ     14$
 1107  005756' 104403  000000' 011344'         MSGN$,BEGIN,RMRR        ;ASCII MESSAGE CALL WITH COMMON HEADER
 1108  005764' 032767  000002  172574  14$:    BIT     #BIT1,S+14              ;ILR ?
 1109  005772' 001403                          BEQ     15$
 1110  005774' 104403  000000' 011350'         MSGN$,BEGIN,ILRR        ;ASCII MESSAGE CALL WITH COMMON HEADER
 1111  006002' 032767  000001  172556  15$:    BIT     #BIT0,S+14              ;ILF ?
 1112  006010' 001403                          BEQ     16$
 1113  006012' 104403  000000' 011354'         MSGN$,BEGIN,ILFR        ;ASCII MESSAGE CALL WITH COMMON HEADER
 1114  006020' 000207                  16$:    RTS     PC
 1115
 1116                                          ;       R E V  D   END OF UPDATE
 1117
 1118
 1119                                          ;////\\\\////\\\\////\\\\////\\\\////\\\\////\\\\////\\\\////\\\\////\\\\
 1120                                          ;////\\\\////\\\\////\\\\////\\\\////\\\\////\\\\////\\\\////\\\\////\\\\
 1121
 1125
 1126
 1127  006022'                         ERRORS:                                                         ;DRB001
 1128  006022' 005777  173752                  TST     @#RHCS1         ;IF NO ERROR THEN               ;DRB001
 1129  006026' 100402                          BMI     1$              ;BEGIN                          ;DRB001
 1130  006030' 000241                          CLC                     ; SET NO ERROR                  ;DRB001
 1131  006032' 000207                          RTS     PC
 1132
 1133  006034'                         1$:                             ;ELSE                           ;DRB001
 1134  006034' 000261                          SEC                     ;SET C BIT TO INDICATE ERROR
 1135  006036' 005000                          CLR     R0              ;INIT INDEX
 1136  006040' 016701  173734                  MOV     RHCS1,R1        ; BEGIN OF REGISTER POINTERS    ;DRB001
 1137  006044'                         3$:                             ; REPEAT                        ;DRB001
 1138                                                                  ; BEGIN                         ;DRB001
 1139  006044' 012160  000552'                 MOV     (R1)+,S(R0)     ;  MOV REGISTER TO TEMPORARY    ;DRB001
 1140  006050' 005720                          TST     (R0)+           ;  ADVANCE POINTER              ;DRB001
 1141  006052' 022760  177777  000552'         CMP     #-1,S(R0)               ;END ?   REV D
 1142  006060' 001401                          BEQ     33$             ;YES END OF TABLE               ;OJM01
 1143  006062' 000770                          BR      3$              ;  NO GO DO IT AGAIN            ;OJM01
 1144  006064' 004767  000470          33$:    JSR     PC,ERSUB1       ; GET SOME TRIVIAL INFO         ;DRB001
 1145
 1146
 1147
 1148  006070' 000167  176426                  JMP     ERDR            ;               REV. DO  ;MCP
 1149
 1150  006074' 032767  000100  171714  BAD:    BIT     #BDSPT,SR1      ;  IF OPERATOR WANTS TYPEOUT THEN       ;DRB001
 1151  006102' 001434                          BEQ     12$             ;                               ;DRB001
 1152  006104'                         11$:                            ;  BEGIN                        ;DRB001
 1153  006104' 116767  172450  172540          MOVB    S+6,BADSEC      ;   SET CURRENT SECTOR          ;DRB001
 1154  006112' 116767  172443  172534          MOVB    S+7,BADTRK      ;   SET CURRENT TRACK           ;DRB001
 1155  006120' 016767  172464  172522          MOV     S+36,BADCYL     ;   SET CURRENT CYLINDER        ;DRB001
 1156                                          ;***************************************************************
  (1)                                         ;CONVERT UNITNO TO ASCII AND
```



```
  (1)                                         ;STORE AT ADR1
  (1)  006126' 104420  000000' 000676'          OTOA$,BEGIN,UNITNO,ADR1
  (1)  006134' 011374'
  (1)                                         ;***************************************************************
 1157                                         ;***************************************************************
  (1)                                         ;CONVERT BADCYL TO ASCII AND
  (1)                                         ;STORE AT CYLNO
  (1)  006136' 104420  000000' 000650'          OTOA$,BEGIN,BADCYL,CYLNO
  (1)  006144' 007472'
  (1)                                         ;***************************************************************
 1158                                         ;***************************************************************
  (1)                                         ;CONVERT BADTRK TO ASCII AND
  (1)                                         ;STORE AT TRKNO
  (1)  006146' 104420  000000' 000654'          OTOA$,BEGIN,BADTRK,TRKNO
  (1)  006154' 007510'
  (1)                                         ;***************************************************************
 1159                                         ;***************************************************************
  (1)                                         ;CONVERT BADSEC TO ASCII AND
  (1)                                         ;STORE AT SECNO
  (1)  006156' 104420  000000' 000652'          OTOA$,BEGIN,BADSEC,SECNO
  (1)  006164' 007526'
  (1)                                         ;***************************************************************
 1160  006166' 104403  000000' 011122'          MSGN$,BEGIN,BADMES      ;ASCII MESSAGE CALL WITH COMMON HEADER
 1161  006174'                         12$:                            ;  END                          ;DRB001
 1162  006174' 026727  172436  000552'          CMP     NXTBBK,#ENDBBK  ;  IF MORE SPACE IN TABLE THEN  ;DRB001
 1163  006202' 002010                          BGE     13$             ;  BEGIN                        ;DRB001
 1164  006204' 016700  172426                  MOV     NXTBBK,R0       ;   R0 POINTS TO NEXT FREE SLOT IN TABLE;DRB001
 1165  006210' 016720  172434                  MOV     BADCYL,(R0)+    ;   STUFF CYLINDER NUMBER       ;DRB001
 1166  006214' 016720  172340                  MOV     S+6,(R0)+       ;   STUFF REST OF DISK ADDRESS  ;DRB001
 1167
 1168  006220' 010067  172412                  MOV     R0,NXTBBK       ;   UPDATE POINTER              ;DRB001
 1169  006224'                         13$:                            ;  END                          ;DRB001
 1170  006224' 000207                          RTS     PC
 1171
 1172
 1173
 1174
 1175
 1176  006226' 012767  077777  172456  NOTRDY: MOV     #77777,CLK      ; SET THE TIMER
 1177  006234'                         4$:
 1178  006234' 032777  004000  173536          BIT     #BIT11,@RHCS1   ;DO I HAVE THE DRIVE DVA?
 1179  006242' 001406                          BEQ     6$              ;NO
 1180  006244' 010046                          MOV     R0,-(SP)
 1181  006246' 012600                          MOV     (SP)+,R0        ;JUST WASTE A LITTLE TIME
 1182  006250' 032777  004000  173522          BIT     #BIT11,@RHCS1           ;STILL GOT IT?
 1183  006256' 001017                          BNE     2$
 1184  006260'                         6$:
  (1)  006260' 104407  000000'                BREAK$,BEGIN            ;TEMPORARY RETURN TO MONITOR....
  (1)  006264' 104407  000000'                BREAK$,BEGIN            ;THEN CONTINUE AT NEXT INSTRUCTION.
 1185  006270' 104407  000000'                BREAK$,BEGIN            ;TEMPORARY RETURN TO MONITOR....
  (1)  006274' 104407  000000'                BREAK$,BEGIN            ;THEN CONTINUE AT NEXT INSTRUCTION.
 1186  006300' 005367  172406                  DEC     CLK             ;COUNT # OF TRIES
 1187  006304' 001353                          BNE     4$              ;NOT DONE YET
 1188  006306' 104403  000000' 011146'          MSGN$,BEGIN,NOT ;ASCII MESSAGE CALL WITH COMMON HEADER
 1189  006314' 000426                          BR      7$              ;COULD NOT GET DRIVE
 1190  006316' 004567  174350          2$:     JSR     R5,CLEAR        ;SET THE CONTROLLER AND DRIVE
```

```
 1191  006322' 004567  000074            JSR     R5,READY        ; IS DRIVE READY ?
 1192  006326' 000434                    BR      1$              ; YES, CONTINUE
 1193  006330' 004767  000224    5$:     JSR     PC,ERSUB1       ; LOAD ERROR INFORMATION
 1194  006334' 005000                    CLR     R0              ;MOVE DRIVE REG INTO TABLE
 1195  006336' 016701  173436            MOV     RHCS1,R1
 1196  006342' 012160  000552'    22$:    MOV     (R1)+,S(R0)
 1197  006346' 005720                    TST     (R0)+
 1198  006350' 022700  000046            CMP     #46,R0
 1199  006354' 001372                    BNE     22$
 1200  006356' 012767  000006  171522    MOV     #6,ERRTYP       ;DRIVE NOT READY
 1201                                     ;*****************************************************************
  (1)  006364' 104405  000000' 001722'    HRDERS,BEGIN,TABLE      ; DRIVE  NOT READY
  (1)                                     ;*****************************************************************
 1202                                                                     ;**-1                  ;DRB001
 1203  006372' 004767  174246    7$:     JSR     PC,RELEASE      ;RELEASE THE DRIVE             ;DRB001
 1204  006376' 004767  176062            JSR     PC,DROP         ; NO, DROP THE DRIVE
 1205  006402' 104403  000000' 011164'    MSGNS,BEGIN,DRP ;ASCII MESSAGE CALL WITH COMMON HEADER
 1206  006410' 016706  171416            MOV     SPOINT,R6
 1207  006414' 000167  173636            JMP     LOOP2
 1208  006420' 000207            1$:     RTS     PC              ;RETURN
 1209                               ;     -------------------------
 1210
 1211
 1212  006422' 016777  172250  173360  READY:  MOV     UNITNO,@RHCS2   ; LOAD UNIT ADDRESS
 1213  006430' 017700  173344            MOV     @RHCS1,R0
 1214  006434' 032700  004000            BIT     #BIT11,R0
 1215  006440' 001433                    BEQ     1$
 1216  006442' 017700  173344            MOV     @RPDS,R0        ; SAVE STATUS IN R0
 1217  006446' 105700                    TSTB    R0              ; DRIVE READY ?
 1218  006450' 100027                    BPL     1$              ; NO
 1219  006452' 032700  000100            BIT     #BIT6,R0        ; VOLUME VALID ?
 1220  006456' 001424                    BEQ     1$              ; NO
 1221  006460' 032700  000400            BIT     #BIT8,R0        ; DRIVE PRESENT ?
 1222  006464' 001421                    BEQ     1$              ; NO
 1223  006466' 032700  004000            BIT     #BIT11,R0       ; WRITE LOCKED ?
 1224  006472' 001016                    BNE     1$              ; YES
 1225  006474' 032700  010000            BIT     #BIT12,R0       ; MEDIUM ON LINE ?
 1226  006500' 001413                    BEQ     1$              ; NO
 1227  006502' 032700  040000            BIT     #BIT14,R0       ; ANY ERRORS ?
 1228  006506' 001010                    BNE     1$              ; YES
 1229  006510' 005700                    TST     R0              ; ATTENTION SET ?
 1230  006512' 100406                    BMI     1$              ; YES
 1231  006514' 032777  004000  173256    BIT     #BIT11,@RHCS1   ;DVA SET?
 1232  006522' 001402                    BEQ     1$              ;BR IF NOT
 1233                                                                     ;**-3                  ;DRB001
 1234  006524' 000261                    SEC                     ; READY                        ;DRB001
 1235  006526' 000401                    BR      2$              ; RETURN                       ;DRB001
 1236  006530' 000241            1$:     CLC                     ; NOT READY                    ;DRB001
 1237  006532' 000205            2$:     RTS     R5              ;RETURN                        ;DRB001
 1238                               ;     -------------------------
 1239
 1240
 1241  006534' 014167  171346  ERSUB2: MOV     -(R1),ASB       ; LOAD THE DATA
 1242  006540' 010167  171336            MOV     R1,SBADR        ; LOAD ADDRESS OF DATA WRITTEN
 1243  006544' 014267  171340            MOV     -(R2),AWAS      ; LOAD THE DATA
 1244  006550' 010267  171330            MOV     R2,WASADR       ; LOAD ADDRESS OF DATA READ
```



```
 1245  006554' 005721                    TST     (R1)+           ; RESET REG. 1
 1246  006556' 005722                    TST     (R2)+           ; RESET REG. 2
 1247
 1248  006560' 016767  173214  171312  ERSUB1: MOV     RHCS1,CSRA      ; LOAD ADR OF CURRENT CSR
 1249  006566' 017767  173206  171306          MOV     @RHCS1,ACSR     ; LOAD CONTENTS OF CURRENT CSR
 1250  006574' 000207                    RTS     PC              ; RETURN
 1251
 1252
 1253  006576' 017700  173176  REZET:  MOV     @RHCS1,R0
 1254  006602' 032700  004000            BIT     #BIT11,R0
 1255  006606' 001005                    BNE     3$
 1256  006610' 104407  000000'           BREAKS,BEGIN            ;TEMPORARY RETURN TO MONITOR....
  (1)  006614' 104407  000000'           BREAKS,BEGIN            ;THEN CONTINUE AT NEXT INSTRUCTION.
 1257  006620' 000766                    BR      REZET
 1258  006622' 012777  000040  173160  3$:     MOV     #BIT5,@RHCS2    ; ISSUE AN RH11 INIT
 1259  006630' 012767  077777  172054          MOV     #77777,CLK      ; SET THE TIMER
 1260  006636' 105777  173136    1$:     TSTB    @RHCS1          ; CONTROLLER READY ?
 1261  006642' 100417                    BMI     2$              ; YES, CONTINUE
 1262  006644' 104407  000000'           BREAKS,BEGIN            ;TEMPORARY RETURN TO MONITOR....
  (1)  006650' 104407  000000'           BREAKS,BEGIN            ;THEN CONTINUE AT NEXT INSTRUCTION.
 1263  006654' 005367  172032            DEC     CLK             ; WAIT SOME MORE ?
 1264  006660' 001366                    BNE     1$              ; YES
 1265  006662' 012767  000003  171216    MOV     #3,ERRTYP       ;CONTROLLER NOT READY
 1266                                     ;*****************************************************************
  (1)  006670' 104405  000000' 000000      HRDERS,BEGIN,NULL       ; CONTROLLER NOT READY
  (1)                                     ;*****************************************************************
 1267  006676' 104410  000000'           ENDS,BEGIN              ;
 1268  006702' 017746  173110    2$:     MOV     @RPAS,-(SP)
 1269  006706' 012677  173104            MOV     (SP)+,@RPAS
 1270  006712' 000207                    RTS     PC              ; RETURN
 1271                               ;     -------------------------
 1272
 1273
 1274  006714' 016700  171066  SETUP:  MOV     ADDR,R0         ; GET DEVICE ADDRESS
 1275  006720' 010067  173054          MOV     R0,RHCS1        ; GENERATE REGISTER ADDRESSES
 1276  006724' 062700  000002          ADD     #2,R0
 1277  006730' 010067  173046          MOV     R0,RHWC
 1278  006734' 062700  000002          ADD     #2,R0
 1279  006740' 010067  173040          MOV     R0,RHBA
 1280  006744' 062700  000002          ADD     #2,R0
 1281  006750' 010067  173032          MOV     R0,RPDA
 1282  006754' 062700  000002          ADD     #2,R0
 1283  006760' 010067  173024          MOV     R0,RHCS2
 1284  006764' 062700  000002          ADD     #2,R0
 1285  006770' 010067  173016          MOV     R0,RPDS
 1286  006774' 062700  000002          ADD     #2,R0
 1287  007000' 010067  173010          MOV     R0,RPER1
 1288  007004' 062700  000002          ADD     #2,R0
 1289  007010' 010067  173002          MOV     R0,RPAS
 1290  007014' 062700  000002          ADD     #2,R0
 1291  007020' 010067  172774          MOV     R0,RPLA
 1292  007024' 062700  000002          ADD     #2,R0
 1293  007030' 010067  172766          MOV     R0,RHDB
 1294  007034' 062700  000002          ADD     #2,R0
 1295  007040' 010067  172760          MOV     R0,RPMR
 1296  007044' 062700  000002          ADD     #2,R0
```

```
1297  007050' 010067 172752            MOV     R0,RPDT
1298  007054' 062700 000002            ADD     #2,R0
1299  007060' 010067 172744            MOV     R0,RPSN
1300  007064' 062700 000002            ADD     #2,R0
1301  007070' 010067 172736            MOV     R0,RPOF
1302  007074' 062700 000002            ADD     #2,R0
1303  007100' 010067 172730            MOV     R0,RPDC
1304  007104' 062700 000002            ADD     #2,R0
1305  007110' 010067 172722            MOV     R0,RPCC
1306  007114' 062700 000002            ADD     #2,R0
1307  007120' 010067 172714            MOV     R0,RPER2
1308  007124' 062700 000002            ADD     #2,R0
1309  007130' 010067 172706            MOV     R0,RPER3
1310  007134' 062700 000002            ADD     #2,R0
1311  007140' 010067 172700            MOV     R0,RPEC1
1312  007144' 062700 000002            ADD     #2,R0
1313  007150' 010067 172672            MOV     R0,RPEC2
1314                                                                ;**-7           ;DRB001
1315  007154' 032767 001000 170674      BIT     #ADDR22,RES1    ;DO WE HAVE 22-BIT ADDR SUPPORT?  ;DRB001
1316  007162' 001415                    BEQ     1$              ;NO               ;DRB001
1317  007164' 032767 000200 170624      BIT     #RH70,SR1       ;IS THIS AN RH70  ;DRB001
1318  007172' 001411                    BEQ     1$              ;NO               ;DRB001
1319  007174' 062700 000002            ADD     #2,R0
1320  007200' 010067 172644            MOV     R0,RHBAE
1321  007204' 062700 000002            ADD     #2,R0
1322  007210' 010067 172636            MOV     R0,RHCS3
1323
1324  007214' 000403                    BR      2$                                ;DRB001
1325  007216' 042767 001000 170632 1$:  BIC     #ADDR22,RES1    ; CAN'T USE 22 BIT ADDRESSING  ;DRB001
1326  007224' 016700 170560      2$:    MOV     VECTOR,R0       ; GET VECTOR ADDRESS
1327  007230' 012720 003104'            MOV     #NTRUPT,(R0)+   ; SET POINTER JUST IN CASE
1328  007234' 116710 170552            MOVB    BR1,(R0)        ; SET PRIORITY
1329  007240' 000207                    RTS     PC              ; RETURN
1330
```



```
1332                                      .SBTTL  MODULE MESSAGES                 ;DRB001
1333                                                                ;**-40          ;DRB001
1334  007242' 042440 051122 051117 MES1:   .ASCIZ  ' ERROR%'                      ;DRB001
      007250' 000045
1335  007252' 050040 051101 052111 MES2:   .ASCIZ  ' PARITY'                       ;DRB001
      007260' 000131
1336  007262' 042040 044522 042526 MES3:   .ASCIZ  ' DRIVE '                       ;DRB001
      007270' 000040
1337  007272' 046440 051501 041123 MES4:   .ASCIZ  ' MASSBUS'                      ;DRB001
      007300' 051525    000
1338  007303'    040 040504 040524 MES5:   .ASCIZ  ' DATA'                        ;DRB001
      007310'    000
1339  007311'    040 042522 051124 MES6:   .ASCIZ  ' RETRY EXCEEDED%'             ;DRB001
      007316' 020131 054105 042503
      007324' 042105 042105 000045
1340  007332' 046040 052101 000105 MES7:   .ASCIZ  ' LATE'                        ;DRB001
1341  007340' 047516 020124 042522 MES8:   .ASCIZ  'NOT READY%'                   ;DRB001
      007346' 042101 022531    000
1342  007353'    040 047503 046125 MES9:   .ASCIZ  ' COULD NOT GET'              ;DRB001
      007360' 020104 047516 020124
      007366' 042507 000124
1343  007372' 042040 047522 050120 MES10:  .ASCIZ  ' DROPPED'                     ;DRB001
      007400' 042105    000
1344  007403'    072 047040 052117 MES11:  .ASCIZ  ': NOT RP04/5/6%'             ;DRB001
      007410' 051040 030120 027464
      007416' 027465 022466    000
1345  007423'    040 051124 047101 MES12:  .ASCIZ  ' TRANSFER'                   ;DRB001
      007430' 043123 051105    000
1346  007435'    045    000        MES13:  .ASCIZ  '%'                          ;DRB001
1347  007437'    072 041040 042101 MES15:  .ASCIZ  ': BADSPOT AT (OCTAL) CYL: '  ;DRB001
      007444' 050123 052117 040440
      007452' 020124 047450 052103
      007460' 046101 020051 054503
      007466' 035114 000040
1348  007472' 000002                    CYLNO:  .BLKB   2                        ;DRB001
1349  007474' 020040 020040 020054 MES16:  .ASCIZ  '    , TRK: '              ;DRB001
      007502' 051124 035113 000040
1350  007510' 000004                    TRKNO:  .BLKB   4                        ;DRB001
1351  007514' 020040 020054 042523 MES17:  .ASCIZ  '  , SEC: '                  ;DRB001
      007522' 035103 000040
1352  007526' 000004                    SECNO:  .BLKB   4                        ;DRB001
1353  007532' 020040 000045     MES18:  .ASCIZ  '  %'                          ;DRB001
1354  007536' 052440 045516 047516 MES19:  .ASCIZ  ' UNKNOWN'                    ;DRB001
      007544' 047127    000
1355
1356
1357
1361                                      ;/////\\\\\/////\\\\\/////\\\\\/////\\\\\/////\\\\\
1362                                      ;\\\\\/////\\\\\/////\\\\\/////\\\\\/////\\\\\/////
1363
1364
1365                                      ;REV D  UPDATES          ;MCP
1366
1367  007547'    110 051101 020104 MES20:  .ASCIZ  'HARD ERROR(S) SET IN RPER2  %'
      007554' 051105 047522 024122
      007562' 024523 051440 052105
```

```
                007570' 044440  020116  050122
                007576' 051105  020062  022440
                007604'    000
 1368           007605'    110  051101  020104  MES21:  .ASCIZ  'HARD ERROR(S) SET IN RPER3  % '
                007612' 051105  047522  024122
                007620' 024523  051440  052105
                007626' 044440  020116  050122
                007634' 051105  020063  022440
                007642' 000040
 1369           007644' 051127  052111  020105  MES22:  .ASCIZ  'WRITE CHECK ERROR  % '
                007652' 044103  041505  020113
                007660' 051105  047522  020122
                007666' 022440  000040
 1370           007672' 041523  051440  052105  MES23:  .ASCIZ  'SC SET  %'
                007700' 020040  000045
 1371           007704' 044124  020105  047506  MES24:  .ASCIZ  'THE FOLLOWING ERROR BITS ARE SET IN RPCS2 % % '
                007712' 046114  053517  047111
                007720' 020107  051105  047522
                007726' 020122  044502  051524
                007734' 040440  042522  051440
                007742' 052105  044440  020116
                007750' 050122  051503  020062
                007756' 020045  020045     000
 1372           007763'    125  044516  052502  MES25:  .ASCIZ  'UNIBUS PARITY ERROR % '
                007770' 020123  040520  044522
                007776' 054524  042440  051122
                010004' 051117  022440  000040
 1373           010012' 047516  020116  054105  MES26:  .ASCIZ  'NON EXISTANT DRIVE % '
                010020' 051511  040524  052116
                010026' 042040  044522  042526
                010034' 022440  000040
 1374           010040' 054116  020115  022440  MES27:  .ASCIZ  'NXM  %'
                010046'    000
 1375           010047'    120  047522  051107  MES28:  .ASCIZ  'PROGRAM ERROR  PGE  % '
                010054' 046501  042440  051122
                010062' 051117  020040  043520
                010070' 020105  022440  000040
 1376           010076' 044515  051523  042105  MES29:  .ASCIZ  'MISSED TRANSFER  %  '
                010104' 052040  040522  051516
                010112' 042506  020122  022440
                010120' 020040     000
 1377           010123'    115  051501  041123  MES30:  .ASCIZ  'MASSBUS DATA LINE PARITY ERROR  % '
                010130' 051525  042040  052101
                010136' 020101  044514  042516
                010144' 050040  051101  052111
                010152' 020131  051105  047522
                010160' 020122  022440  000040
 1378           010166' 044124  020105  047506  MES31:  .ASCIZ  'THE FOLLOWING BITS ARE SET IN RPER1  % % '
                010174' 046114  053517  047111
                010202' 020107  044502  051524
                010210' 040440  042522  051440
                010216' 052105  044440  020116
                010224' 050122  051105  020061
                010232' 022440  022440  000040
 1379           010240' 040504  040524  041440  MES32:  .ASCIZ  'DATA CHECK ERROR  % '
                010246' 042510  045503  042440
```



```
                010254' 051122  051117  020040
                010262' 020045     000
 1380           010265'    106  051117  040515  MES33:  .ASCIZ  'FORMAT ERROR  %  '
                010272' 020124  051105  047522
                010300' 020122  022440  020040
                010306'    000
 1381           010307'    040  042510  042101  MES34:  .ASCIZ  ' HEADER CRC ERROR % '
                010314' 051105  041440  041522
                010322' 042440  051122  051117
                010330' 022440  000040
 1382           010334' 042510  042101  051105  MES35:  .ASCIZ  'HEADER COMPARE ERROR  % '
                010342' 041440  046517  040520
                010350' 042522  042440  051122
                010356' 051117  020040  020045
                010364'    000
 1383           010365'    104  044522  042526  MES36:  .ASCIZ  'DRIVE UNSAFE  % '
                010372' 052440  051516  043101
                010400' 020105  022440  000040
 1384           010406' 050117  051105  052101  MES37:  .ASCIZ  'OPERATION INCOMPLETE % '
                010414' 047511  020116  047111
                010422' 047503  050115  042514
                010430' 042524  022440  000040
 1385           010436' 051104  053111  020105  MES38:  .ASCIZ  'DRIVE TIMING ERROR % '
                010444' 044524  044515  043516
                010452' 042440  051122  051117
                010460' 022440  000040
 1386           010464' 051127  052111  020105  MES39:  .ASCIZ  'WRITE LOCK ERROR % '
                010472' 047514  045503  042440
                010500' 051122  051117  022440
                010506' 000040
 1387           010510' 047111  040526  044514  MES40:  .ASCIZ  'INVALID ADDRESS ERROR % '
                010516' 020104  042101  051104
                010524' 051505  020123  051105
                010532' 047522  020122  020045
                010540'    000
 1388           010541'    105  041503  044040  MES41:  .ASCIZ  'ECC HARD ERROR % '
                010546' 051101  020104  051105
                010554' 047522  020122  020045
                010562'    000
 1389           010563'    120  051101  052111  MES42:  .ASCIZ  'PARITY ERROR % '
                010570' 020131  051105  047522
                010576' 020122  020045     000
 1390           010603'    122  043505  051511  MES43:  .ASCIZ  'REGISTER MODIFICATION REFUSED  % '
                010610' 042524  020122  047515
                010616' 044504  044506  040503
                010624' 044524  047117  051040
                010632' 043105  051525  042105
                010640' 020040  020045     000
 1391           010645'    111  046114  043505  MES44:  .ASCIZ  'ILLEGAL REGISTER % '
                010652' 046101  051040  043505
                010660' 051511  042524  020122
                010666' 020045     000
 1392           010671'    111  046114  043505  MES45:  .ASCIZ  'ILLEGAL FUNCTION % '
                010676' 046101  043040  047125
                010704' 052103  047511  020116
                010712' 020045     000
```

```
1393  010715'  040  022445  020045  MES46:  .ASCIZ  ' %%%  ********************************** %% '
      010722' 020040  025052  025052
      010730' 025052  025052  025052
      010736' 025052  025052  025052
      010744' 025052  025052  025052
      010752' 025052  025052  025052
      010760' 025052  025052  025052
      010766' 025052  022440  020045
      010774'  000
1394  010775'  101  042104  042522  MES47:  .ASCIZ  'ADDRESS OVERFLOW ERROR % '
      011002' 051523  047440  042526
      011010' 043122  047514  020127
      011016' 051105  047522  020122
      011024' 020045  000
1395  011027'  127  044522  042524  MES48:  .ASCIZ  'wRITE LOCK FAILURE % '
      011034' 046040  041517  020113
      011042' 040506  046111  051125
      011050' 020105  020045  000
1396
1397
1398                                  ; R E V  D   END OF  UPDATE
1399
1400                                  ;/////\\\\\/////\\\\\/////\\\\\/////\\\\\/////\\\\\
1401                                  ;\\\\\/////\\\\\/////\\\\\/////\\\\\/////\\\\\/////
1402
1403
1407          011056'                 .EVEN                                                ;DRB001
1408                                                                                       ;DRB001
1409  011056' 007423'                 TMERR:  MES12                                        ;DRB001
1410  011060' 007242'                         MES1                                         ;DRB001
1411  011062' 177777                          177777                                       ;DRB001
1412                                                                                       ;DRB001
1413  011064' 007272'                 MCPERR: MES4                                         ;DRB001
1414  011066' 007252'                         MES2                                         ;DRB001
1415  011070' 007242'                         MES1                                         ;DRB001
1416  011072' 177777                          177777                                       ;DRB001
1417                                                                                       ;DRB001
1418  011074' 007272'                 MDPERR: MES4                                         ;DRB001
1419  011076' 007303'                         MES5                                         ;DRB001
1420  011100' 007252'                         MES2                                         ;DRB001
1421  011102' 007242'                         MES1                                         ;DRB001
1422  011104' 177777                          177777                                       ;DRB001
1423                                                                                       ;DRB001
1424  011106' 007311'                 EXCED:  MES6                                         ;DRB001
1425  011110' 177777                          177777                                       ;DRB001
1426                                                                                       ;DRB001
1427  011112' 007303'                 DLTERR: MES5                                         ;DRB001
1428  011114' 007332'                         MES7                                         ;DRB001
1429  011116' 007242'                         MES1                                         ;DRB001
1430  011120' 177777                          177777                                       ;DRB001
1431                                                                                       ;DRB001
1432  011122' 007262'                 BADMES: MES3                                         ;DRB001
1433  011124' 011401'                         NUMb                                         ;DRB001
1434  011126' 007437'                         MES15                                        ;DRB001
1435  011130' 007474'                         MES16                                        ;DRB001
1436  011132' 007514'                         MES17                                        ;DRB001
```



```
1437  011134' 007532'                         MES18                                        ;DRB001
1438  011136' 177777                          177777                                       ;DRB001
1439                                                                                       ;DRB001
1440  011140' 007536'                 UNKNWN: MES19                                        ;DRB001
1441  011142' 007242'                         MES1                                         ;DRB001
1442  011144' 177777                          177777                                       ;DRB001
1443                                                                                       ;DRB001
1444  011146' 007262'                 NOT:    MES3                                         ;DRB001
1445  011150' 007340'                         MES8                                         ;DRB001
1446  011152' 177777                          177777                                       ;DRB001
1447                                                                                       ;DRB001
1448  011154' 007353'                 TOUT:   MES9                                         ;DRB001
1449  011156' 007262'                         MES3                                         ;DRB001
1450  011160' 007435'                         MES13                                        ;DRB001
1451  011162' 177777                          177777                                       ;DRB001
1452                                                                                       ;DRB001
1453  011164' 007262'                 DRP:    MES3                                         ;DRB001
1454  011166' 011401'                         NUMB                                         ;DRB001
1455  011170' 007372'                         MES10                                        ;DRB001
1456  011172' 007435'                         MES13                                        ;DRB001
1457  011174' 177777                          177777                                       ;DRB001
1458                                                                                       ;DRB001
1459  011176' 007262'                 BADTYP: MES3                                         ;DRB001
1460  011200' 011401'                         NUMB                                         ;DRB001
1461  011202' 007372'                         MES10                                        ;DRB001
1462  011204' 007403'                         MES11                                        ;DRB001
1463  011206' 177777                          177777                                       ;DRB001
1464
1465
1469                                  ;/////\\\\\/////\\\\\/////\\\\\/////\\\\\/////\\\\\
1470                                  ;\\\\\/////\\\\\/////\\\\\/////\\\\\/////\\\\\/////
1471
1472                                  ;R E V  D  UPDATES
1473
1474  011210' 007547'                 ER2ERR: MES20
1475  011212' 177777                          177777
1476
1477  011214' 007605'                 ER3ERR: MES21
1478  011216' 177777                          177777
1479
1480  011220' 007644'                 WCEERR: MES22
1481  011222' 177777                          177777
1482
1483  011224' 007672'                 SCERR:  MES23
1484  011226' 177777                          177777
1485
1486  011230' 007704'                 CS2HD:  MES24
1487  011232' 177777                          177777
1488
1489  011234' 007763'                 UPER:   MES25
1490  011236' 177777                          177777
1491
1492  011240' 010012'                 NEDER:  MES26
1493  011242' 177777                          177777
1494
1495  011244' 010040'                 NXM:    MES27
```

```
1496  011246' 177777                  177777
1497
1498  011250' 010047'           PGERR:  MES28
1499  011252' 177777                  177777
1500
1501  011254' 010076'           MXFER:  MES29
1502  011256' 177777                  177777
1503
1504  011260' 010123'           MDPER:  MES30
1505  011262' 177777                  177777
1506
1507  011264' 010166'           ER1HD:  MES31
1508  011266' 177777                  177777
1509
1510  011270' 010240'           DCKER:  MES32
1511  011272' 177777                  177777
1512
1513  011274' 010265'           FERER:  MES33
1514  011276' 177777                  177777
1515
1516  011300' 010307'           HCRC:   MES34
1517  011302' 177777                  177777
1518
1519  011304' 010334'           HCE:    MES35
1520  011306' 177777                  177777
1521
1522  011310' 010365'           UNSER:  MES36
1523  011312' 177777                  177777
1524
1525  011314' 010406'           OPIER:  MES37
1526  011316' 177777                  177777
1527
1528  011320' 010436'           DTER:   MES38
1529  011322' 177777                  177777
1530  011324' 010464'           WLER:   MES39
1531  011326' 177777                  177777
1532  011330' 010510'           IAER:   MES40
1533  011332' 177777                  177777
1534
1535  011334' 010541'           ECHR:   MES41
1536  011336' 177777                  177777
1537
1538  011340' 010563'           PARR:   MES42
1539  011342' 177777                  177777
1540
1541  011344' 010603'           RMRR:   MES43
1542  011346' 177777                  177777
1543
1544  011350' 010645'           ILRR:   MES44
1545  011352' 177777                  177777
1546
1547  011354' 010671'           ILFR:   MES45
1548  011356' 177777                  177777
1549  011360' 010715'           ASTER:  MES46
1550  011362' 177777                  177777
1551
```



```
1552  011364' 010775'           AOE:    MES47                   ;       REV. D0
1553  011366' 177777                   177777
1554
1555  011370' 011027'           WCF:    MES48                   ;       REV. D0
1556  011372' 177777                   177777
1557
1558
1559
1560                                    ;R E V  D  END UPDATE
1561
1562                                    ;////\\\\////\\\\////\\\\////\\\\////\\\\
1563                                    ;////\\\\////\\\\////\\\\////\\\\////\\\\
1564
1565                                                                            ;DRB001
1566  011374' 000005            ADR1:   .BLKB   5                               ;DRB001
1567  011401'    040     000    NUMB:   .ASCIZ  / /                             ;DRB001
1568          011404'                   .EVEN                                   ;DRB001
1569                                                                            ;DRB001
1570                                    ;-----------------------------          ;DRB001
1571                                                                            ;DRB001
1572          000001                    .END                                    ;DRB001
```

```
ACSR    000102R          370#   1249#
ADDR    000006R          370#   1274
ADDR22= 001000           370#    615    1315    1325
ADR1    011374R          934    1156    1566#
ADE     011364R         1094    1552#
ASB     000106R          370#   1241#
ASTAT   000104R          370#
ASTER   011360R          954    1016    1549#
AWAS    000110R          370#   1243#
BAD     006074R          652    1101    1150#
BADCYL  000650R          410#   1155#   1157    1165
BADMES  011122R         1160    1432#
BADSEC  000652R          411#   1153#   1159
BADSPT  000252R          373#    401     489     782
BADTRK  000654R          412#   1154#   1158
BADTYP  011176R          901    1459#
BDSPT = 000100           386#   1150
BEGIN   000000R          370#    520     523     535     536     545     596     622     630     638     651     705     901
                         934     954     963     967     979     983     987     991     995    1007    1009    1016    1023
                        1026    1029    1032    1035    1038    1042    1045    1049    1052    1063    1066    1079    1082
                        1085    1088    1091    1094    1097    1100    1104    1107    1110    1113    1156    1157    1158
                        1159    1160    1184    1185    1188    1201    1205    1256    1262    1266    1267
BITTAB  003310R          633     671     683#
BIT0  = 000001           370#    758     867     877     884     894    1111
BIT1  = 000002           370#   1108
BIT10 = 002000           370#   1033    1089
BIT11 = 004000           370#   1030    1086    1178    1182    1214    1223    1231    1254
BIT12 = 010000           370#    611    1027    1083    1225
BIT13 = 020000           370#    964    1024    1080
BIT14 = 040000           370#    610     971     984     988    1077    1227
BIT15 = 100000           370#    956     992    1046    1055
BIT2  = 000004           370#    383    1105
BIT3  = 000010           370#   1102
BIT4  = 000020           370#    384    1050    1055
BIT5  = 000040           370#    385    1095    1258
BIT6  = 000100           370#    386    1098    1219
BIT7  = 000200           370#    387    1064
BIT8  = 000400           370#   1040    1060    1221
BIT9  = 001000           370#   1036    1092
BPORT = 000020           384#    756
BREAK$= 104407           370#    596    1184    1185    1256    1262
BR1     000012R          370#   1328
BR2     000013R          370#
BTOD$ = 104421           370#
CDATA$= 104412           370#    545
CDERCT  000144R          370#
CDWDCT  000146R          370#
CHKCS2  005156R          970    1022#
CHKER1  005320R          975    1045#
CLEAR   002672R          600#   1190
CLK     000712R          430#   1176#   1186#   1259#   1263#
CLRCOM  004454R          922#    924
CLRRB   004444R          920#
CONFIG  000056R          370#
CSRA    000100R          370#   1248#
CS2HD   011230R         1023    1486#
```



```
CYCLE   002322R          530    539#
CYL     000644R          405#   510#    708#    735#    741    745#    749    752#    755    767#
CYLLIM  000640R          403#   741     750     760     868#   878#   885#   895#
CYLNDR  000642R          404#   571     581     591     755#   760#   762    764#    786    802
CYLNO   007472R         1157   1348#
DATCK$= 104411           370#
DATER$= 104404           370#
DCKER   011270R         1049   1510#
DLATE = 000004           383#    960
DLTCNT  000700R          425#   958#
DLTERR  011112R          963   1427#
DLTFLG  005154R          953#   959#    986#    990#    994#   1002    1004   1019#   1048#   1058#   1062#
DRIVE   000674R          422#
DROP    004464R          507    900     927#   1204
DRP     011164R         1205   1453#
DSKADR  000646R          406#   511#    570     580     590     746#    768#
DTER    011320R         1085   1528#
DVICE   000672R          421#   494#    504     533     671     933#
DVID1   000014R          370#   494
EA22    000634R          399#   624     625#    626#    627
ECHR    011334R         1100   1535#
ENDBBK  000552R          381#   1162
ENDIT$= 104413           370#   536
END$  = 104410           370#   535    1267
EROR    004522R          953#   1148
ERRORS  006022R          640    1127#
ERRTYP  000106R          370#   1200#   1265#
ERSUB1  006560R         1144    1193    1248#
ERSUB2  006534R         1241#
ER1HD   011264R         1045   1507#
ER2ERR  011210R          979   1474#
ER3ERR  011214R          983   1477#
EXCED   011106R          651   1424#
EXITS = 104400           370#   630
FERADR  000710R          429#   565#    575#    585#    650
FERER   011274R         1052   1513#
FLAG    000670R          420#   758     867#    877#    884#    894#
FREE    000150R          370#
FUNC    000702R          426#   564#    574#    584#    628#    629
GETDVT  004202R          679    859#
GETPA$= 104415           370#   520
GO      002756R          573    583     593     614#
GWBUF$= 104414           370#   523
HCE     011304R         1066   1519#
HCRC    011300R         1063   1516#
HRDCNT  000044R          370#
HRDCS2= 137400           388#    968
HRDERR  000720R          433#   645     955#    962#    966#    978#    982#   1000   1022#   1057#   1071#   1076#
HRDER$= 104405           370#   1009    1201    1266
HRDER1= 177777           389#    973
HRDER2= 167777           390#    976
HRDER3= 140153           391#    980
HRDPAS  000050R          370#
IAER    011330R         1091   1532#
ICONT   000036R          370#
ICOUNT  000040R          370#
```

```
IDNUM   000122R          370#
ILFR    011354R         1113    1547#
ILRR    011350R         1110    1544#
IMODX.= 000000           370#    523
INIT    000030R          370#
INTR    000120R          370#    488#
LASCYL  000656R          414#    762     764     854#    869#    879#    886#    896#
LASSEC  000662R          416#    773     775     857#
LASTRK  000660R          415#    770     772     856#
LOOP1   002244R          522#    537
LOOP2   002256R          525#    531     657    1207
MAP22S= 104416           370#    622
MCPERR  011064R          967    1413#
MDPER   011260R         1042    1504#
MDPERR  011074R         1418#
MES1    007242R         1334#   1410    1415    1421    1429    1441
MES10   007372R         1343#   1455    1461
MES11   007403R         1344#   1462
MES12   007423R         1345#   1409
MES13   007435R         1346#   1450    1456
MES15   007437R         1347#   1434
MES16   007474R         1349#   1435
MES17   007514R         1351#   1436
MES18   007532R         1353#   1437
MES19   007536R         1354#   1440
MES2    007252R         1335#   1414    1420
MES20   007547R         1367#   1474
MES21   007605R         1368#   1477
MES22   007644R         1369#   1480
MES23   007672R         1370#   1483
MES24   007704R         1371#   1486
MES25   007763R         1372#   1489
MES26   010012R         1373#   1492
MES27   010040R         1374#   1495
MES28   010047R         1375#   1498
MES29   010076R         1376#   1501
MES3    007262R         1336#   1432    1444    1449    1453    1459
MES30   010123R         1377#   1504
MES31   010166R         1378#   1507
MES32   010240R         1379#   1510
MES33   010265R         1380#   1513
MES34   010307R         1381#   1516
MES35   010334R         1382#   1519
MES36   010365R         1383#   1522
MES37   010406R         1384#   1525
MES38   010436R         1385#   1528
MES39   010464R         1386#   1530
MES4    007272R         1337#   1413    1418
MES40   010510R         1387#   1532
MES41   010541R         1388#   1535
MES42   010563R         1389#   1538
MES43   010603R         1390#   1541
MES44   010645R         1391#   1544
MES45   010671R         1392#   1547
MES46   010715R         1393#   1549
MES47   010775R         1394#   1552
```



```
MES48   011027R         1395#   1555
MES5    007303R         1338#   1419    1427
MES6    007311R         1339#   1424
MES7    007332R         1340#   1428
MES8    007340R         1341#   1445
MES9    007353R         1342#   1448
MODLUS  004432R          729     738     751     909#
MODNAM  000000R          370#
MODSP   000252R          370#
MSGN$ = 104403           370#    651     901     954     963     967     979     983     987     991     995    1016    1023
                        1026    1029    1032    1035    1038    1042    1045    1049    1052    1063    1066    1079    1082
                        1085    1088    1091    1094    1097    1100    1104    1107    1110    1113    1160    1188    1205
MSGS$ = 104402           370#
MSG$  = 104401           370#
MXFER   011254R         1038    1501#
NEDER   011240R         1029    1492#
NORAND= 000040           385#    698     743     765
NOT     011146R         1188    1444#
NOTRDY  006226R          678    1013    1176#
NTRUPT  003104R          631#   1327
NULL  = 000000           370#   1266
NUMB    011401R         1433    1454    1460    1567#
NXM     011244R         1032    1495#
NXTBBK  000636R          401#    493#    783    1162    1164    1168#
OPEN  = 000000           370#
OPIER   011314R         1082    1525#
OTOA$ = 104420           370#    934    1156    1157    1158    1159
PARR    011340R         1104    1538#
PASCNT  000034R          370#
PA18    000626R          396#    617#    622
PA22    000632R          398#    623
PGERR   011250R         1035    1498#
PICKBK  003320R          541     655     689#    814
PICKDR  003226R          526     664#
PIRQ$ = 000004           370#    638
POPSP = 005726           370#
POPSP2= 022626           370#
PRTY  = 000000           370#
PRTY0 = 000000           370#
PRTY1 = 000040           370#
PRTY2 = 000100           370#
PRTY3 = 000140           370#
PRTY4 = 000200           370#
PRTY5 = 000240           370#
PRTY6 = 000300           370#
PRTY7 = 000340           370#
PS    = 177776           370#
PS.#  = 177776           370#
PUSH  = 005746           370#
PUSH2 = 024646           370#
RAND$ = 104417           370#    705
RANNUM  000054R          370#    706     709     713
RPDP    000722R          370     434#    920
RPDPEA  000110R          370#    542
RPDPPA  000126R          370#    545     589
RPDPSZ  000132R          370#    586     921
```

```
RBUFVA  000124R          370#    520
READ    002550R          544     584#    585
READY   006422R          676    1191    1212#
RELESE  002644R          546     594#    1203
RESTRT  002236R          370     516     518#
RES1    000056R          370#    615    1315    1325*
RES2    000060R          370#
REZET   006576R          514    1253#   1257
RHBA    002004R          463#    569*    579*    589*    617     623*    1279*
RHBAE   002050R          481#    624*   1320*
RHCS1   002000R          461#    595*    601*    604*    605     610*    629*    964    1128    1136    1178    1182    1195
                        1213    1231    1248    1249    1253    1260    1275*
RHCS2   002010R          465#    594*    600*    614*    864*   1212*   1258*   1283*
RHCS3   002052R          482#   1322*
RHDB    002022R          470#    998     999*   1010    1012*   1293*
RHWC    002002R          462#    568*    578*    588*   1277*
RH70  = 000200           387#    515    1317
RMRR    011344R         1107    1541#
RPAS    002016R          468#    608     609*    634     636*   1268    1269*   1289*
RPCC    002036R          476#   1305*
RPDA    002006R          464#    570*    580*    590*   1281*
RPDC    002034R          475#    571*    581*    591*   1303*
RPDS    002012R          466#   1216    1285*
RPDT    002026R          472#    865     872     874     882     889     891    1297*
RPEC1   002044R          479#   1311*
RPEC2   002046R          480#   1313*
RPER1   002014R          467#   1287*
RPER2   002040R          477#   1307*
RPER3   002042R          478#   1309*
RPLA    002020R          469#   1291*
RPMR    002024R          471#   1295*
RPOF    002032R          474#    611*   1301*
RPSN    002030R          473#   1299*
RSTRT   000112R          370#
S       000552R          393#    437     438     439     440     441     442     443     444     445     446     447     448
                         449     450     451     452     453     454     455     456     457     458     517*    956     968
                         971     973     976     980     984     988     992     996    1024    1027    1030    1033    1036
                        1040    1046    1050    1053    1060    1064    1067    1072    1077    1080    1083    1086    1089
                        1092    1095    1098    1102    1105    1108    1111    1139*   1141    1153    1154    1155    1166
                        1196*
SBADR   000102R          370#   1242*
SCERR   011224R          995    1483#
SECNO   007526R         1159    1352#
SECTOR  000646R          407#    700*    712*    724     727     730*    773     775*    784     810
SETUP   006714R          513    1274#
SIZSEC  000666R          419#    788     841*
SIZTRK  000664R          418#    794     835*    838*    850
SOFCNT  000042R          370#
SOFERS= 104406           370#   1007
SOFPAS  000046R          370#
SPOINT  000032R          370#   1206
SPSIZ = 000040             7#    370
SR1     000016R          370#    515     698     743     756     765     960    1150    1317
SR2     000020R          370#
SR3     000022R          370#
SR4     000024R          370#
```



```
START   002054R          370     486#
STAT    000026R          370#
STORE   000716R          432#    998#    1012    1053#   1054#   1055    1067#   1068#   1069
SVR0    000062R          370#
SVR1    000064R          370#
SVR2    000066R          370#
SVR3    000070R          370#
SVR4    000072R          370#
SVR5    000074R          370#
SVR6    000076R          370#
SYSCNT  000052R          370#
TABLE   001722R          436#   1007    1009    1201
TIMER   000704R          427#
TOUT    011154R         1448#
TRACK   000647R          408#    722#    726#    733     736     739#    770     772#    785     806
TRERR   011056R          991    1409#
TRKNO   007510R         1158    1350#
TRPDFD= 000022           370#
TRY     000714R          431#    529#    647#    648
UNITNU  000676R          423#    506#    522#    594     600     614     632     656#    665#    666     670     864     928
                         934    1156    1212
UNKNWN  011140R         1440#
UNSER   011310R         1079    1522#
UPER    011234R         1026    1489#
VECTOR  000010R          370#   1326
WASADR  000104R          370#   1244#
WBUFEA  000136R          370#    572     582
WBUFPA  000134R          370#    569     579
WBUFRQ  000140R          370#
WBUFSZ  000142R          370#    566     576     832
WCEERR  011220R          987    1480#
WCF     011370R         1097    1555#
WDFR    000116R          370#    487#
WDTO    000114R          370#    486#
WLER    011324R         1088    1530#
WRITCK  002454R          543     574#    575
WRITE   002360R          542     564#    565
WRTLIM  004062R          528     822#
XFLAG   000005R          370#
XMEM    000630R          397#    572#    582#    592#    618#    619#    620#    621#    627#    628
ZERO    000706R          428#
.     = 011404R          434#    545    1348#   1350#   1352#   1407#   1566#   1568#
```

```
BKMOD      95#
BREAK     185#    596    1184    1185    1256    1262
BTOD      204#
CKDATA    240#    545
DATACK    249#
DATERR    138#
DFSEVN    272#    370
DSEVNT    262#    370
END       175#    535    1267
ENDIT     166#    536
ENDMOD    171#
EQUATS    288#    370
EXIT      120#    630
GETPA     231#    520
GWBUFF    219#    523
HRDER     128#   1009    1201    1266
IOMOD      91#
IOMODP    115#
IOMODR    111#
IOMODX    107#    370
LINEUP    551#    572     582     592
MAP22     235#    622
MODULE      8#    370
MSG       154#
MSGN      158#    901     954     963     967     979     983     987     991     995    1016    1023    1026    1029    1032
         1035    1038    1042    1045    1049    1052    1063    1066    1079    1082    1085    1088    1091    1094    1097
         1100    1104    1107    1110    1113    1160    1188    1205
MSGS      162#
NBKMOD    103#
OTOA      190#    934    1156    1157    1158    1159
PIRQ      179#    638
RAND      124#    705
SBKMOD     99#
SOFER     144#   1007


. ABS.  000000      000
        011404      001

 ERRORS DETECTED:  0
 DEFAULT GLOBALS GENERATED:  0

 XRPDD0.OBJ,XRPDD0.LST/CRF=DDXCOM.P11,XRPDD0.P11
 RUN-TIME: 6 12 1 SECONDS
 RUN-TIME RATIO: 45/22=2.0
 CORE USED:  8K  (15 PAGES)
```