RP11-C/RP02

MULTIDRIVE

MD-11-DZRPG-B

EP-DZRPG-B-DL-A OCT 1976

digital

Made in U.S.A.

FICHE 1 OF 1

[illegible]

.REM %

IDENTIFICATION

PRODUCT CODE: MAINDEC-11-DZRPG-B-D

PRODUCT NAME: RP11 MULTI DRIVE DIAGNOSTIC

DATE CREATED: JUNE 15, 1973

MAINTAINER: DIAGNOSTIC GROUP

AUTHOR: JOE STUBBLEBINE

CONTENTS

1. ABSTRACT

THIS PROGRAM WILL TEST UP TO EIGHT RP02 DRIVES ON AN RP11 DISK CONTROLLER. BASICALLY, THE PROGRAM WILL SEEK TO A RANDOM ADDRESS AND THEN WRITE AND READ RANDOM DATA. WHILE DATA IS BEING TRANSFERRED, SEEK OPERATIONS WILL BE IN PROGRESS ON THE OTHER DRIVES. THE PURPOSE OF THE TEST IS TO CHECK FOR ANY INTERACTION ON THE BUS WHILE TRYING TO KEEP ALL THE DRIVES BUSY.

2. REQUIREMENTS

2.1 EQUIPMENT

PDP-11 STANDARD FAMILY PROCESSOR
RP11 DISK CONTROLLER WITH UP TO EIGHT RP02 DISK DRIVES

2.2 STORAGE

4K OF STORAGE IS REQUIRED TO RUN THIS TEST

2.3 PRELIMINARY PROGRAMS

DZRPE DISKLESS DIAGNOSTIC
DZRPF DISK RELIABILITY DIAGNOSTIC

3. LOADING PROCEDURE

USE STANDARD PROCEDURE FOR ABS TAPES.

4. STARTING PROCEDURE

4.1 CONTROL SWITCH SETTINGS

SEE 5.1.1

```
   144
   145                                  4.2     STARTING ADDRESS
```

THE PROGRAM SHOULD ALWAYS BE STARTED AT 200.

4.3 PROGRAM AND/OR OPERATOR ACTION

1. LOAD PROGRAM INTO MEMORY USING ABS LOADER
2. LOAD ADDRESS 200
3. SET SWITCHES (SEE SEC. 5.1.1)
4. PRESS START.
5. THE PROGRAM WILL LOOP UNTIL STOPPED
6. DUE TO THE RANDOM NATURE OF THE PROGRAM THERE IS NO MEANINGFULL PASSCOUNT. IT IS RECOMMENDED THAT THE PROGRAM RUN AT LEAST HALF AN HOUR.

5. OPERATING PROCEDURE

5.1 OPERATIONAL SWITCH SETTINGS

AT STARTING ADDRESS 200, THE SETTING OF THE SWITCHES WILL DETERMINE WHICH UNITS ARE TO BE TESTED.

5.1.1 SWITCH SETTING ARE:

```
SW<15>=1.....HALF ON ERROR
SW<14>  .....NOT USED
SW<13>=1.....INHIBIT PRINTOUT
SW<12>  .....NOT USED
SW<11>  .....NOT USED
SW<10>=1.....BELL ON ERROR
SW<07> THRU SW<00>=1.....SELECT UNIT FOR TEST

SW<00> CORRESPONDS TO UNIT 0
SW<07> CORRESPONDS TO UNIT 7
```

5.2 SUBROUTINE ABSTRACTS

5.2.1 HLT

```
202
203                                 THIS ROUTINE IS ENTERED UPON DETECTION OF AN  ERROR.
```

IT WILL TYPE THE PC OF THE ERROR AND ADDITIONAL ERROR INFORMATION. THIS ROUTINE TESTS FOR, HALT ON ERROR, INHIBIT TYPEOUTS, AND RINGS THE BELL.

5.2.2 TRAP CATCHER

A ".+2" - "HALT" SEQUENCE IS REPEATED FROM 0-776 TO CATCH ANY UNEXPECTED TRAPS. THESE UNEXPECTED TRAPS OR INTERRUPTS WILL HALT AT THE VECTOR +2.

6.0 ERRORS

6.1 WHEN ERRORS ARE ENCOUNTERED, THE ADDRESS OF THE ERROR ALONG WITH THE CONTENTS OF RPDS, RPER, AND RPCS ARE TYPED. ALSO, THE CONTENTS OF THE SELECTED CYLINDER, HEAD AND SECTOR ADDRESS ARE TYPED. BY REFERRING TO THE LISTING, ADDITIONAL INFORMATION CAN BE FOUND REGARDING THE CAUSE OF THE ERROR IN THE COMMENT FIELD. WHEN APPROPRIATE, ADDITIONAL INFORMATION IS TYPED OUT, SUCH AS THE EXPECTED AND RECEIVED RESULTS OF AN OPERATION. ALL INFORMATION IS IN OCTAL.

ERROR MESSAGE FORMAT

1. PC= ADDRESS OF FAILURE
   UNIT UNIT WHICH FAILED
   RPDS= CONTENTS OF RPDS
   RPER= CONTENTS OF RPER
   RPCS= CONTENTS OF RPCS
   CYLINDER SELECTED CYLINDER
   HEAD SELECTED HEAD
   SECTOR SELECTED SECTOR
   EXPECTED EXPECTED DATA
   RECEIVED RECEIVED DATA

7.0 RESTRICTIONS

SINCE THIS IS AN INTERACTION TEST, THERE IS NO LOOPING ON ERRORS.

8.0 MISCELLANEOUS

8.1 EXECUTION TIME

DUE TO THE RANDOM NATURE OF THE PROGRAM THERE IS NO MEANINGFUL PASS COUNT. IT IS RECOMMENDED THAT THE PROGRAM SHOULD RUN FOR HALF AN HOUR.

8.2 STACK POINTER

STACK IS INITIALLY SET TO 500.

8.3 ERROR INFORMATION

IF IT IS DESIRED TO HAVE THE ERROR INFORMATION OUTPUTTED TO THE PUNCH INSTEAD OF THE TELETYPE CHANGE THE FOLLOWING THREE LOCATIONS.

LOCATION	FROM	TO
1304	177564	177554
1332	177566	177556
1336	177564	177554

9.0 PROGRAM DESCRIPTION

WHEN STARTED THE PROGRAM WILL RESTORE THE HEADS FOR EACH OF THE SELECTED UNITS. THEN THE FOLLOWING SEQUENCE IS GONE THRU FOR EACH OF THE SELECTED UNITS. FIRST, A RANDOM DISK SURFACE ADDRESS IS GENERATED AND A SEEK IS ISSUED. THEN A RANDOM BUFFER IS SELECTED AND FILLED WITH RANDOM DATA. A SECTOR IS THEN WRITTEN, READ BACK AND COMPARED. THIS SEQUENCE IS THEN LOOPED UPON. DUE TO THE DIFFERENCE IN SEEK TIMES, WHICH DEPENDS ON THE RANDOM DISK ADDRESS SELECTED, ALL UNITS ARE EXERCISED IN A RANDOM SELECTION. WHILE DATA IS BEING TRANSFERRED, SEEK OPERATIONS ARE IN PROGRESS.

%

```
000000' 012706 000000G        START:  MOV     #STKPTR,SP      ;SET STACK POINTER
000004' 012737 000340 000000G         MOV     #340,@#PSW      ;LOCK UP INTERRUPTS
000012' 012767 000000G 000034         MOV     #ERROR,34       ;SETUP ERROR TRAP
000020' 012767 000000G 000036         MOV     #PRI7,36
000026' 012767 000000G 000030         MOV     #SCOPE$,30      ;SETUP SCOPE TRAP
```

```
   315  000034' 012767  000000G 000032          MOV     #PRI7,32
   316  000042' 012737  002336' 000254          MOV     #DSKINT,@#VECTOR        ;SET UP DISK INTERRUPT VECTOR
   317  000050' 012737  000340  000256          MOV     #340,@#STATUS
   318  000056' 005000                          CLR     R0
   319  000060' 005060  002632'         CLRTAB: CLR     DEVTBL(R0)      ;CLEAR THE DEVICE TABLE
   320  000064' 005720                          TST     (R0)+
   321  000066' 022700  000200                  CMP     #128.,R0
   322  000072' 001372                          BNE     CLRTAB
   323  000074' 005737  000042                  TST     @#42            ;UNDER MONITOR CONTROL?
   324  000100' 001424                          BEQ     5$              ;BRANCH IF NO
   325  000102' 005000                          CLR     R0              ;CLEAR MODIFIER
   326  000104' 005001                          CLR     R1
   327  000106' 012777  000001  002742  7$:     MOV     #1,@RPCS        ;CLEAR THE CONTROLLER
   328  000114' 110177  002740                  MOVB    R1,@RPCS1       ;SELECT UNIT
   329  000120' 005777  002750                  TST     @RPDS           ;IS UNIT READY?
   330  000124' 100403                          BMI     6$              ;BRANCH IF YES
   331  000126' 052760  000000G 002632'        BIS     #B15,DEVTBL(R0) ;SET UNIT UNAVAILABLE BIT
   332  000134' 062700  000020          6$:     ADD     #16.,R0         ;UPDATE MODIFIER
   333  000140' 005201                          INC     R1              ;UPDATE UNIT NUMBER
   334  000142' 032701  000000G                 BIT     #B3,R1          ;ALL UNITS TESTED?
   335  000146' 001757                          BEQ     7$              ;BRANCH IF NO
   336  000150' 000420                          BR      8$
   337  000152' 012701  000001          5$:     MOV     #1,R1
   338  000156' 005000                          CLR     R0
   339  000160' 030137  000000G         2$:     BIT     R1,@#SWR        ;TEST THE SWITCH REGISTER
   340  000164' 001003                          BNE     1$              ;TO DETERMINE WHICH UNITS
   341  000166' 052760  000000G 002632'        BIS     #B15,DEVTBL(R0) ;TO TEST. IF THE UNIT IS UNAVAILABLE
   342  000174' 062700  000020          1$:     ADD     #16.,R0         ;SET BIT 15 IN THE DEVICE TABLE
   343  000200' 000241                          CLC
   344  000202' 006101                          ROL     R1
   345  000204' 032701  000000G                 BIT     #B8,R1          ;HAVE ALL UNITS BEEN SCANNED?
   346  000210' 001763                          BEQ     2$              ;NO-BRANCH
   347  000212' 000005                  8$:     RESET                   ;CLEAR THE SYSTEM
   348  000214' 004567  001656                  JSR     R5,EXTMEN       ;DETERMINE AMOUNT OF CORE
   349  000220' 005067  002606                  CLR     UNIT            ;INITIALIZE POINTER
   350  000224' 005067  002604                  CLR     PASSCT
   351  000230' 005005                          CLR     R5
   352  000232' 032765  000000G 002632' 4$:     BIT     #B15,DEVTBL(R5) ;IS UNIT AVAILABLE?
   353  000240' 001002                          BNE     3$              ;BRANCH IF NO
   354  000242' 004767  000136                  JSR     PC,HOME         ;DO A HOME SEEK
   355  000246' 005267  002560          3$:     INC     UNIT            ;UPDATE UNIT
   356  000252' 062705  000020                  ADD     #16.,R5         ;UPDATE TABLE POINTER
   357  000256' 032767  000000G 002546         BIT     #B3,UNIT        ;HAVE ALL UNITS BEEN HOMED?
   358  000264' 001762                          BEQ     4$              ;NO-BRANCH
   359  000266' 005067  002540          LOOP:   CLR     UNIT
   360  000272' 005005                          CLR     R5
   361  000274' 032765  000000G 002632' MAIN:   BIT     #B15,DEVTBL(R5) ;IS THE UNIT AVAILABLE?
   362  000302' 001004                          BNE     1$              ;BRANCH IF NO
   363  000304' 016504  002632'                 MOV     DEVTBL(R5),R4
   364  000310' 004774  000550'                 JSR     PC,@JMPTBL(R4)  ;PERFORM FUNCTION IN JMPTBL
   365  000314' 005267  002512          1$:     INC     UNIT            ;UPDATE UNIT
   366  000320' 062705  000020                  ADD     #16.,R5         ;UPDATE TABLE POINTER
   367  000324' 032767  000000G 002500         BIT     #B3,UNIT        ;HAVE ALL UNITS BEEN SCANNED?
   368  000332' 001760                          BEQ     MAIN            ;NO BRANCH
   369  000334' 005267  002474                  INC     PASSCT          ;INCREMENT ITERATION COUNTER
   370  000340' 016737  002470  000000G         MOV     PASSCT,@#SWR    ;DISPLAY COUNT
```

```
   371  000346' 005737  000042                  TST     @#42            ;UNDER MONITOR CONTROL?
   372  000352' 001413                          BEQ     MEXIT1          ;BRANCH IF NO
   373  000354' 022767  005000  002452          CMP     #5000,PASSCT    ;IS PASS COMPLETE?
   374  000362' 001007                          BNE     MEXIT1          ;BRANCH IF NO
   375  000364' 013701  000042                  MOV     @#42,R1         ;GET RETURN ADDRESS
   376  000370' 000005                          RESET
   377  000372' 004711                  MEXIT:  JSR     PC,(R1)         ;RETURN TO MONITOR
   378  000374' 000240                          NOP
   379  000376' 000240                          NOP
   380  000400' 000240                          NOP
   381  000402' 000731                  MEXIT1: BR      LOOP            ;LOOP
   382
   383                                          ;THIS ROUTINE WILL SEEK HOME THE PACK WHOSE
   384                                          ;ADDRESS IS IN UNIT.
   385
   386  000404' 116777  002422  002446  HOME:   MOVB    UNIT,@RPCS1     ;LOAD THE UNIT #
   387  000412' 005777  002456                  TST     @RPDS           ;IS THE UNIT READY?
   388  000416' 100401                          BMI     5$              ;BRANCH IF READY
   389  000420' 000000G                         HLT                     ;UNIT IS NOT READY
   390  000422' 112777  000015  002426  5$:     MOVB    #15,@RPCS       ;DO A HOME SEEK
   391  000430' 012704  000025                  MOV     #25,R4
   392  000434' 005304                  1$:     DEC     R4              ;WAIT FOR SEEK TO START
   393  000436' 001376                          BNE     1$
   394  000440' 032777  000000G 002426          BIT     #B10,@RPDS      ;IS SEEK UNDER WAY SET
   395  000446' 001001                          BNE     2$              ;YES
   396  000450' 000000G                         HLT                     ;SEEK UNDERWAY DID NOT SET
   397  000452' 016704  002354          2$:     MOV     UNIT,R4
   398  000456' 005067  000054                  CLR     ATTNB
   399  000462' 116467  000540' 000046          MOVB    ATTN(R4),ATTNB  ;DERTMINE ATTENTION RESPONSE
   400  000470' 005000                          CLR     R0
   401  000472' 005200                  7$:     INC     R0
   402  000474' 001376                          BNE     7$
   403  000476' 005200                  6$:     INC     R0              ;TIME OUT ATTENTION BIT
   404  000500' 036777  000032  002366          BIT     ATTNB,@RPDS     ;DID ATTENTION SET?
   405  000506' 001003                          BNE     3$              ;BRANCH IF YES
   406  000510' 005700                          TST     R0              ;DID IT TIME OUT?
   407  000512' 001371                          BNE     6$              ;BRANCH IF NO
   408  000514' 000000G                         HLT                     ;ATTENTION BIT DID NOT SET
   409
   410  000516' 005777  002334          3$:     TST     @RPCS           ;ANY DEVICE STATUS ERRORS?
   411  000522' 100001                          BPL     4$              ;NO-BRANCH
   412  000524' 000000G                         HLT                     ;DEVICE STATUS ERROR AFTER HOME COMMAND
   413  000526' 046777  000004  002340  4$:     BIC     ATTNB,@RPDS     ;CLEAR ATTENTION BIT
   414  000534' 000207                          RTS     PC              ;EXIT
   415
   416  000536' 000000                  ATTNB:  0                       ;CONTAINS ATTENTION BIT FOR CURRENT UNIT
   417  000540'    001     002     004  ATTN:   .BYTE   1,2,4,10,20,40,100,200
   418  000543'    010     020     040
   419  000546'    100     200
   420
   421                                          .EVEN
   422                                  ;THIS TABLE DETERMINES WHERE CONTROL WILL GO FOR ANY
   423                                  ;PARTICULAR UNIT.  THE FIRST WORD OF THE DEVICE TABLE
   424                                  ;IS USED AS A MODIFIER FOR A JSR INDIRECT INTO
   425                                  ;THE FOLLOWING TABLE.
   426
```

```
   427  000550' 000560'                 JMPTBL: SEEK                    ;SEEK A RANDOM CYLINDER
   428  000552' 001040'                         SEKCK                   ;SEE IF SEEK IS COMPLETE
   429  000554' 001354'                         WRITE                   ;WRITE RANDOM DATA
   430  000556' 001530'                         READD                   ;READ AND VERIFY RANDOM DATA
   431
   432                                  ;THIS ROUTINE WILL GENERATE A RANDOM CYLINDER
   433                                  ;ADDRESS AND ISSUE A SEEK TO IT.  THE FUNCTION
   434                                  ;POINTER IN THE DEVICE TABLE WILL BE UPDATED TO
   435                                  ;CHECK FOR THE ATTENTION BIT.
   436
   437  000560' 016565  002636' 002634' SEEK:   MOV     DEVTBL+4(R5),DEVTBL+2(R5) ;SET UP CYLINDER FROM ADDRESS
   438  000566' 116777  002240  002264          MOVB    UNIT,@RPCS1     ;SELECT THE UNIT
   439  000574' 000000G                 1$:     RAND                    ;GENERATE A RANDOM NUMBER
   440  000576' 016767  000000G 002234         MOV     LONUM,WORK1
   441  000604' 016767  000000G 002230         MOV     HINUM,WORK2
   442  000612' 042767  177400  002220         BIC     #177400,WORK1
   443  000620' 022767  000312  002212         CMP     #312,WORK1      ;IS NUMBER TOO LARGE?
   444  000626' 002762                         BLT     1$              ;BRANCH IF YES
   445  000630' 026765  002204  002634'        CMP     WORK1,DEVTBL+2(R5)
   446  000636' 001756                         BEQ     1$
   447  000640' 016765  002174  002636'        MOV     WORK1,DEVTBL+4(R5)      ;SAVE CYLINDER ADDRESS
   448  000646' 016765  002170  002640'        MOV     WORK2,DEVTBL+6(R5)      ;USE A RANDOM DATA BASE
   449  000654' 000000G                         RAND                    ;GENERATE A RANDOM NUMBER
   450  000656' 016767  000000G 002154         MOV     LONUM,WORK1
   451  000664' 016767  000000G 002150         MOV     HINUM,WORK2
   452  000672' 042767  177760  002140         BIC     #177760,WORK1
   453  000700' 022767  000011  002132         CMP     #11,WORK1       ;GENERATE A RANDOM SECTOR
   454  000706' 002003                         BGE     2$
   455  000710' 162767  000010  002122         SUB     #8.,WORK1
   456  000716' 042767  177740  002116  2$:    BIC     #177740,WORK2   ;GENERATE A RANDOM TRACK
   457  000724' 022767  000023  002110         CMP     #23,WORK2
   458  000732' 002003                         BGE     3$
   459  000734' 162767  000014  002100         SUB     #14,WORK2
   460  000742' 116767  002074  002071  3$:    MOVB    WORK2,WORK1+1   ;MERGE TRACK AND SECTOR ADDR
   461
   462  000750' 016765  002064  002642'        MOV     WORK1,DEVTBL+10(R5)     ;STORE RANDOM DISK ADDRESS
   463  000756' 005065  002644'                CLR     DEVTBL+12(R5)   ;CLEAR TIMEOUT COUNTER
   464  000762' 116577  002636' 002076         MOVB    DEVTBL+4(R5),@RPCA      ;LOAD CYLINDER ADDRESS
   465  000770' 016577  002642' 002072         MOV     DEVTBL+10(R5),@RPDA     ;LOAD TRACK AND SECTOR
   466  000776' 112777  000011  002052         MOVB    #11,@RPCS       ;INITIATE A SEEK
   467  001004' 012704  000025                 MOV     #25,R4
   468  001010' 005304                  4$:    DEC     R4              ;WAIT FOR SEEK TO START
   469  001012' 001376                         BNE     4$
   470  001014' 032777  000000G 002052         BIT     #810,@RPDS      ;IS THE SEEK UNDERWAY?
   471  001022' 001001                         BNE     5$              ;YES-BRANCH
   472  001024' 000000G                         HLT                     ;SEEK UNDERWAY DID NOT SET
   473  001026' 005265  002632'         5$:    INC     DEVTBL(R5)      ;MODIFY FUNCTION POINTER TO
   474  001032' 005265  002632'                 INC     DEVTBL(R5)      ;CHECK FOR SEEK COMPLETE
   475  001036' 000207                         RTS     PC              ;EXIT
   476
   477                                  ;THIS ROUTINE IS ENTERED AFTER A SEEK HAS BEEN ISSUED.
   478                                  ;IT CHECKS THE ATTENTION FLAG - IF CLEAR IT UPDATES THE
   479                                  ;TIMEOUT COUNTER AND CHECKS IT.  IF SET IT VERIFIES
   480                                  ;THE SEEK FUNCTIONED PROPERLY.
   481
   482  001040' 016704  001766         SEKCK:  MOV     UNIT,R4
```

```
                                                MOVB    ATTN(R4),ATTNB  ;DETERMINE ATTENTION BIT
                                                BIT     ATTNB,@RPDS     ;TEST FOR ATTENTION FLAG
                                                BNE     1$              ;BRANCH IF SET
                                                INC     DEVTBL+12(R5)   ;UPDATE TIMEOUT COUNTER
                                                CMP     #5000,DEVTBL+12(R5)     ;DID OPERATION TIMEOUT?
                                                BHI     2$              ;NOT YET-BRANCH
                                                MOVB    UNIT,@RPCS1     ;SELECT UNIT
                                                HLT                     ;SEEK TIMED OUT
                                                BR      4$
                                        2$:     RTS     PC              ;EXIT
                                        1$:     BICB    ATTNB,@RPDS     ;CLEAR ATTENTION FLAG
                                                MOVB    UNIT,@RPCS1     ;SELECT UNIT
                                                BIT     #B10,@RPDS      ;IS SEEK UNDERWAY CLEAR?
                                                BEQ     3$              ;IF YES-BRANCH
                                                HLT                     ;SEEK UNDERWAY DID NOT CLEAR
                                                BR      4$
                                        3$:     TST     @RPCS           ;ARE THERE ANY DEVICE STATUS ERRORS?
                                                BPL     5$              ;BRANCH-NO ERRORS
                                                HLT                     ;DEVICE STATUS ERRORS
                                                BR      4$
                                        5$:     MOVB    @RPCA1,REC$     ;GET CURRENT CYLENDER ADDRESS
                                                CMP     REC$,DEVTBL+4(R5)       ;DOES IT MATCH CYLINDER REQUESTED?
                                                BEQ     6$              ;YES-BRANCH
                                                HLT                     ;CURRENT CYLINDER AND CYL REQUESTED DID NOT COMPARE
                                                PRINT
                                                MES10
Q                                               SDUMP   DEVTBL+4(R5)
                                                PRINT
                                                MES11
Q                                               DUMP    REC$
                                        4$:     BIT     #B11,@RPDS      ;SEEK INCOMPLETE?
                                                BEQ     10$             ;BRANCH IF NO
                                                MOVB    #13,@RPCS       ;ISSUE HOME COMMAND
                                                TSTB    @RPCS           ;WAIT FOR DONE
                                                BPL     .-4
                                                TST     @RPDS           ;WAIT FOR UNIT READY
                                                BPL     .-4
                                        10$:    CLR     DEVTBL(R5)      ;CLEAR FUNCTION POINTER
                                                RTS     PC              ;EXIT
                                        6$:     INC     DEVTBL(R5)
                                                INC     DEVTBL(R5)      ;UPDATE FUNCTION POINTER
                                                RTS     PC              ;EXIT

001262' 005015  042522  052521  MES10:  .ASCIZ  <15><12>'REQUESTED CYLINDER= '
001270' 051505  042524  020104
001276' 054503  044514  042116
001304' 051105  020075     000
001311'    015  041412  051125  MES11:  .ASCIZ  <15><12>'CURRENT CYLINDER ADDR REGISTER= '
001316' 042522  052116  041440
001324' 046131  047111  042504
001332' 020122  042101  051104
001340' 051040  043505  05151[
001346' 042524  036522  000040
                                                .EVEN

                                        ;THIS ROUTINE WILL WRITE A RANDOM SECTOR ON
```

```
                                        ;A RANDOM TRACK.  THE CYLINDER HAS ALREADY
                                        ;BEEN SELECYED BY THE SEEK ROUTINE.

001354' 004767  000634          WRITE:  JSR     PC,RANADR       ;GENERATE A RANDOM BUFFER ADDR
001360' 012767  000400  001450          MOV     #400,WORK
001366' 016701  001456                  MOV     BUFF,R1
001372' 004767  000334                  JSR     PC,RANDAT       ;GENERATE A RANDOM PATTERN
001376' 116777  001430  001454          MOVB    UNIT,@RPCS1     ;SELECT THE UNIT
001404' 032777  000000G 001462          BIT     #B15,@RPDS      ;IS THE SELECTED UNIT READY
001412' 001003                          BNE     1$              ;YES-BRANCH
001414' 000000G                         HLT                     ;SELECTED UNIT NOT READY
001416' 000167  000100                  JMP     WRTER           ;START SEQUENCE OVER

001422' 012777  177400  001432  1$:     MOV     #-400,@RPWC     ;SETUP WORD COUNT REGISTER
001430' 016777  001414  001426          MOV     BUFF,@RPBA      ;SETUP BUS ADDR REGISTER
001436' 016577  002636' 001422          MOV     DEVTBL+4(R5),@RPCA      ;SET CYLINDER ADDR
001444' 016577  002642' 001416          MOV     DEVTBL+10(R5),@RPDA     ;SETUP DISK ADDR.
001452' 005067  001374                  CLR     INT             ;CLEAR INTERRUPT FLAG
001456' 012737  000000G 000000G         MOV     #PRI4,@#PSW     ;ALLOW INTERRUPT
001464' 005067  001356                  CLR     INTERR          ;CLEAR ERROR FLAG
001470' 112777  000113  001360          MOVB    #113,@RPCS      ;INITIATE WRITE WITH INTERRUPT
001476' 004767  000606                  JSR     PC,WAIT         ;TIMEOUT THE OPERATION
001502' 005767  001340                  TST     INTERR          ;ANY ERRORS?
001506' 001005                          BNE     WRTER           ;BRANCH IF YES
001510' 005265  002632'                 INC     DEVTBL(R5)      ;UPDATE FUNCTION POINTER TO READ
001514' 005265  002632'                 INC     DEVTBL(R5)
001520' 000403                          BR      READD
001522' 005065  002632'         WRTER:  CLR     DEVTBL(R5)      ;RESTORE FUNCTION POINTER
001526' 000207                          RTS     PC              ;EXIT

                                        ;READ AND VERIFY THE DATA WRITTEN

001530' 116777  001276  001322  READD:  MOVB    UNIT,@RPCS1     ;SELECT THE UNIT
001536' 032777  000000G 001330          BIT     #B15,@RPDS      ;IS THE SELECTED UNIT READY?
001544' 001003                          BNE     1$              ;YES-BRANCH
001546' 000000G                         HLT                     ;SELECTED UNIT NOT READY
001550' 000167  000150                  JMP     RDCNT
001554' 012777  177400  001300  1$:     MOV     #-400,@RPWC     ;LOAD WORD COUNT REGISTER
001562' 016777  001262  001274          MOV     BUFF,@RPBA      ;LOAD BUS ADDR REGISTER
001570' 062777  001000  001266          ADD     #1000,@RPBA
001576' 016577  002636' 001262          MOV     DEVTBL+4(R5),@RPCA      ;SET CYLINDER ADDR
001604' 016577  002642' 001256          MOV     DEVTBL+10(R5),@RPDA     ;SETUP DISK ADDR.
001612' 005067  001234                  CLR     INT             ;CLEAR INTERRUPT FLAG
001616' 005067  001224                  CLR     INTERR          ;CLEAR ERROR FLAG
001622' 112777  000117  001226          MOVB    #117,@RPCS      ;INITIATE READ WITH INTERRUPT
001630' 004767  000454                  JSR     PC,WAIT         ;TIMEOUT THE OPERATION
001634' 032777  000000G 001214          BIT     #B14,@RPCS      ;ANY HARD ERRORS?
001642' 001030                          BNE     RDCNT           ;BRANCH IF YES
001644' 016701  001200                  MOV     BUFF,R1
001650' 016702  001174                  MOV     BUFF,R2
001654' 005003                          CLR     R3
001656' 062701  001000                  ADD     #1000,R1        ;START OF PATTERN BUFFER
001662' 022122                  3$:     CMP     (R1)+,(R2)+     ;COMPARE DATA
001664' 001006                          BNE     2$              ;BRANCH-DATA DID NOT COMPARE
001666' 005203                          INC     R3              ;INCREMENT COUNTER
001670' 022703  000400                  CMP     #400,R3         ;HAS BUFFER BEEN SCANNED
```

```
                                        BNE     3$                      ;BRANCH-NC
001676' 000167  000022                  JMP     RDCNT
                
001702' 016267  177776  000716  2$:     MOV     -2(R2),EXP$
001710' 016167  177776  000712          MOV     -2(R1),REC$
001716' 0000006                         PRINT
001720' 003205'                         MES12
001722' 0000016                         HLT     +1                      ;DATA DID NOT VERIFY
001724' 005065  002632'         RDCNT:  CLR     DEVTBL(R5)              ;INITIATE FUNCTION POINTER
001730' 000207                          RTS     PC                      ;EXIT

                                ;GENERATE A RANDOM PATTERN

001732' 016567  002640' 000132  RANDAT: MOV     DEVTBL+6(R5),RAND1      ;GET RANDOM BASE
001740' 016567  002642' 000126          MOV     DEVTBL+10(R5),RAND2
001746' 016700  000120                  MOV     RAND1,R0
001752' 016704  000116                  MOV     RAND2,R4
001756' 012703  000007          RANDA1: MOV     #7,R3                   ;SETUP SHIFT COUNT
001762' 005002                  SHIFT:  CLR     R2
001764' 006300                          ASL     R0                      ;SHIFT R0 LEFT AND
001766' 006104                          ROL     R4                      ;ROTATE CARRY INTO LSB OF R0 INTO R4
001770' 006102                          ROL     R2                      ;ROTATE CARRY OUT OF R4 INTO R2
001772' 005303                          DEC     R3                      ;DECREMENT R3
001774' 001373                          BNE     SHIFT                   ;CONTINUE LOOP
001776' 066700  000070                  ADD     RAND1,R0                ;ADD IN # TO MAKE X129
002002' 005504                          ADC     R4                      ;PROPOGATE CARRY
002004' 066704  000064                  ADD     RAND2,R4                ;ADD IN # TO MAKE X129
002010' 005502                          ADC     R2                      ;PROPOGATE CARRY
002012' 062700  001057                  ADD     #1057,R0                ;ADD LOW CONSTANT
002016' 005504                          ADC     R4                      ;PROPOGATE CARRY
002020' 005502                          ADC     R2                      ;PROPOGATE AGAIN
002022' 062704  047401                  ADD     #47401,R4               ;ADD HIGH CONSTANT
002026' 005502                          ADC     R2
002030' 062702  000006                  ADD     #6,R2
002034' 060200                          ADD     R2,R0                   ;REPRIME R0 WITH HIGH DIGIT
002036' 005504                          ADC     R4
002040' 010067  000026                  MOV     R0,RAND1
002044' 010021                          MOV     R0,(R1)+                ;STORE DATA IN BUFFER
002046' 005367  000764                  DEC     WORK
002052' 001406                          BEQ     EXGEN
002054' 010467  000014                  MOV     R4,RAND2
002060' 010421                          MOV     R4,(R1)+                ;STORE DATA IN BUFFER
002062' 005367  000750                  DEC     WORK
002066' 001333                          BNE     RANDA1                  ;FILL ENTIRE BUFFER
002070' 000207                  EXGEN:  RTS     PC                      ;EXIT

002072' 000000                  RAND1:  0
002074' 000000                  RAND2:  0

                                ;THIS ROUTINE DETERMINES THE TOTAL AMOUNT OF AVAILABLE
                                ;CARE WITHOUT USING MEMORY MANAGEMENT.

002076' 012737  0000006 0000006 EXTMEN: MOV     #PRI7,@#PSW             ;LOCKUP PRIORITY LEVEL
002104' 012767  002154' 000004          MOV     #MAXREF,4               ;SETUP IO BUS TRAP
002112' 012767  0000006 000006          MOV     #PRI7,6
002120' 012767  017446  000064          MOV     #17446,SAVE             ;START WITH 4K
```

```
                                EXREF:  TST     @SAVE           ;REFERENCE MEMORY
002126' 005777  000060                  CMP     #157446,SAVE    ;TEST FOR 28K
002132' 022767  157446  000052          BNE     1$              ;BRANCH IF LESS THAN 28K
002140' 001001                          BR      MAXRF1
002142' 000407
002144' 062767  020000  000040  1$:     ADD     #20000,SAVE     ;SETUP FOR NEXT REFERENCE
002152' 000765                          BR      EXREF

                                ;ENTER HERE WHEN IO BUS ERROR OCCURS

002154' 162767  020000  000030  MAXREF: SUB     #20000,SAVE
002162' 012767  000006  000004  MAXRF1: MOV     #6,4            ;RESTORE IO BUS TRAY
002170' 005067  000006                  CLR     6
002174' 005737  000042                  TST     @#42            ;UNDER MONITOR CONTROL?
002200' 001403                          BEQ     1$              ;BRANCH IF NO
002202' 162767  005670  000002          SUB     #3000.,SAVE     ;ADJUST TOP OF CORE
002210' 000205                  1$:     RTS     R5              ;EXIT-SAVE=MAXIMUM MEMORY
002212' 000000                  SAVE:   0                       ;HIGHEST AVAILABLE LOCATION

                                ;GENERATE A RANDOM BUFFER ADDRESS

002214' 016704  177772          RANADR: MOV     SAVE,R4
002220' 162704  003312'                 SUB     #ENDP,R4        ;DETERMINE BUFFER SIZE
002224' 162704  002000                  SUB     #2000,R4        ;ALLOW ROOM FOR DATA
002230' 0000006                         RAND                    ;GENERATE A RANDOM #
002232' 016767  0000006 000600          MOV     LONUM,WORK1
002240' 042767  0000006 000572          BIC     #80,WORK1       ;MAKE NUMBER EVEN
002246' 012703  100000                  MOV     #100000,R3
002252' 020467  000562          2$:     CMP     R4,WORK1        ;ENSURE THAT THE RANDOM
002256' 101005                          BHI     1$              ;ADDRESS FITS WITHIN AVAILABLE
002260' 040367  000554                  BIC     R3,WORK1        ;MEMORY
002264' 000241                          CLC
002266' 006003                          ROR     R3
002270' 000770                          BR      2$
002272' 062767  003312' 000540  1$:     ADD     #ENDP,WORK1
002300' 016767  000534  000542          MOV     WORK1,BUFF      ;SAVE BUFFER START ADDR.
002306' 000207                          RTS     PC              ;EXIT
                                ;TIMEOUT THE OCCURANCE OF AN INTERRUPT

002310' 005000                  WAIT:   CLR     R0
002312' 005200                  2$:     INC     R0
002314' 005767  000532                  TST     INT             ;HAS INTERRUPT OCCURED?
002320' 001005                          BNE     1$              ;YES-BRANCH
002322' 005700                          TST     R0              ;HAS OPERATION TIMED OUT?
002324' 001372                          BNE     2$              ;NO-BRANCH
002326' 0000006                         HLT                     ;UNIT TIMED OUT ON READ OR WRITE
002330' 005267  000512                  INC     INTERR          ;SET ERROR FLAG
002334' 000207                  1$:     RTS     PC              ;EXIT

                                ;ENTERED UPON A DEVICE INTERRUPT.  THIS ROUTINE
                                ;WILL CHECK FOR AND REPORT DEVICE ERRORS

002336' 032777  0000006 000512  DSKINT: BIT     #B15,@RPCS      ;WHERE THER ANY ERRORS?
002344' 001402                          BEQ     1$              ;BRANCH-NO ERRORS
002346' 000167  000064                  JMP     DSKER           ;REPORT ERROR
002352' 016703  000472          1$:     MOV     BUFF,R3
002356' 062703  001000                  ADD     #1000,R3
```

```
                707  002362' 022765  000004  002632'          CMP     #4,DEVTBL(R5)   ;IS THIS A WRITE?
                708  002370' 001402                           BEQ     3$              ;BRANCH IF YES
                709  002372' 062703  001000                   ADD     #1000,R3
                710  002376' 020377  000462           3$:     CMP     R3,@RPBA        ;DID THE BUS ADDR TERMINATE PROPERLY?
                711  002402' 001413                           BEQ     2$              ;YES-BRANCH
                712  002404' 000000G                          HLT                     ;CONTENTS OF RPBA INCORRECT
                713  002406' 000000G                          PRINT
                714  002410' 003232'                          MES13
                715  002412' 000000G                          PRINT
                716  002414' 003262'                          MES18
G               717  002416' 000000G                          DUMP    R3
                718  002420' 000000G                          PRINT
                719  002422' 003276'                          MES19
G               720  002424' 000000G                          DUMP    @RPBA
                721  002426' 005267  000414                   INC     INTERR          ;SET ERROR FLAG
                722  002432' 000167  000006           2$:     JMP     EXTINT
                723  002436' 000000G                  DSKER:  HLT                     ;REPORT INTERRUPT DISK ERROR
                724  002440' 005267  000402                   INC     INTERR          ;SET ERROR FLAG
                725  002444' 052767  000000G 000400  EXTINT: BIS     #80,INT
                726  002452' 032777  000000G 000414  1$:     BIT     #B15,@RPDS      ;IS THE UNIT READY
                727  002460' 001774                           BEQ     1$              ;NO-WAIT
                728  002462' 000002                           RTI                     ;EXIT INTERRUPT
                729
                730
                731  002464' 032767  000000G 000000G MSG:    BIT     #81,HLTCT$      ;TYPE ENTIRE MSG?
                732  002472' 001041                           BNE     1$              ;BRANCH IF NO
                733  002474' 000000G                          PRINT
                734  002476' 003100'                          MES1
G               735  002500' 000000G                          SDUMP   UNIT
                736  002502' 000000G                          PRINT
                737  002504' 003132'                          MES2A
                738  002506' 017767  000352  000000G         MOV     @RPDS,TTY
                739  002514' 004767  000000G                  JSR     PC,PRINTR
                740  002520' 000000G                          PRINT
                741  002522' 003110'                          MES1A
G               742  002524' 000000G                          DUMP    @RPER
                743  002526' 000000G                          PRINT
                744  002530' 003121'                          MES2
G               745  002532' 000000G                          DUMP    @RPCS
                746  002534' 000000G                          PRINT
                747  002536' 003143'                          MES3
G               748  002540' 000000G                          SDUMP   DEVTBL+4(R5)
                749  002542' 000000G                          PRINT
                750  002544' 003160'                          MES4
                751  002546' 005067  000272                   CLR     WORK3
                752  002552' 116567  002643' 000264          MOVB    DEVTBL+11(R5),WORK3
G               753  002560' 000000G                          SDUMP   WORK3
                754  002562' 000000G                          PRINT
                755  002564' 003172'                          MES5
                756  002566' 116567  002642' 000250          MOVB    DEVTBL+10(R5),WORK3
G               757  002574' 000000G                          SDUMP   WORK3
                758  002576' 032767  000000G 000000G 1$:     BIT     #B0,HLTCT$      ;TYPE EXPECTED - RECEIVED?
                759  002604' 001001                           BNE     2$              ;BRANCH IF YES
                760  002606' 000207                           RTS     PC
                761  002610' 000000G                  2$:     PRINT
                762  002612' 003262'                          MES18
```

```
Q  763  002614' 000000G                         DUMP    EXP$
   764  002616' 000000G                         PRINT
   765  002620' 003276'                         MES'9
Q  766  002622' 000000G                         DUMP    REC$
   767  002624' 000207                          RTS     PC
   769  002626' 000000                  EXP$:   0
   769  002630' 000000                  REC$:   0
   770                                  ;DEVTBL IS A TABLE CONTAINING SLOTS FOR EACH OF EIGHT
   771                                  ;POSSIBLE UNITS.  DURING THE OPERATION OF THE PROGRAM
   772                                  ;R5 IS USED AS A MODIFIER TO POINT INTO THE TABLE TO
   773                                  ;SELECT THE PROPER UNIT.  EACH UNIT SLOT CONTAINS
   774                                  ;EIGHT ENTRIES(WORD)
   775                                  ;1 FUNCTION POINTER
   776                                  ;       0=SEEK
   777                                  ;       2=SEEK IN PROGRESS
   778                                  ;       4=WRITE
   779                                  ;       6=READ
   780                                  ;       IF NEGATIVE-UNIT IS NOT TESTED
   781                                  ;2 CYLINDER FROM ADDRESS-INDICATES PREVIOUS CYLINDER POSITION
   782                                  ;3 CYLINDER TO ADDRESS-ADDRESS OF THE SEEK COMMAND
   783                                  ;4 RANDOM BASE FOR PATTERN GENERATION
   784                                  ;5 RANDOM TRACK AND SECTOR ADDRESS
   785                                  ;6 CYLINDER SEEK TIMEOUT COUNTER
   786                                  ;7 SPARE
   787                                  ;8 SPARE
   788
   789                                          .EVEN
   790  002632' 000000          DEVTBL: 0                       ;UNIT 0 SLOT
   791          002652'                 .=DEVTBL+20
   792  002652' 000000          UNIT1:  0                       ;UNIT 1 SLOT
   793          002672'                 .=UNIT1+20
   794  002672' 000000          UNIT2:  0                       ;UNIT 2 SLOT
   795          002712'                 .=UNIT2+20
   796  002712' 000000          UNIT3:  0                       ;UNIT 3 SLOT
   797          002732'                 .=UNIT3+20
   798  002732' 000000          UNIT4:  0                       ;UNIT 4 SLOT
   799          002752'                 .=UNIT4+20
   800  002752' 000000          UNIT5:  0                       ;UNIT 5 SLOT
   801          002772'                 .=UNIT5+20
   802  002772' 000000          UNIT6:  0                       ;UNIT6 SLOT
   803          003012'                 .=UNIT6+20
   804  003012' 000000          UNIT7:  0                       ;UNIT 7 SLOT
   805          003032'                 .=UNIT7+20
   806
   807                                  ;RP11 CONSTANTS-MEMORY ASSIGNMENTS
   808  003032' 000000          UNIT:   0                       ;CURRENT UNTI UNDER TEST
   809  003034' 000000          PASSCT: 0                       ;COUNTS EACH PASS THRU 8 UNITS
   810  003036' 000000          WORK:   0                       ;TEMPORARY STORAGE AREA
   811  003040' 000000          WORK1:  0
   812  003042' 000000          WORK2:  0
   813  003044' 000000          WORK3:  0
   814  003046' 000000          INTERR: 0                       ;INTERRUPT ERROR FLAG
   815  003050' 000000          BUFF:   0                       ;STARTING ADDRESS OF BUFFER
   816  003052' 000000          INT:    0                       ;INTERRUPT FLAG
   817  003054' 000000          FLAG:   0                       ;FLAG WORD
   818                                  ;DISK IO REGISTERS
```

```
                                RPCS:   176714                  ;DISK CONTROL REGISTER
                                RPCS1:  176715
                                RPWC:   176716                  ;DISK WORD COUNT REGISTER
                                RPBA:   176720                  ;CURRENT ADDRSS REGISTER
                                RPCA:   176722                  ;CYLINDER ADDRESS REGISTER
                                RPDA:   176724                  ;DISK ADDRESS REGISTER
                                RPER:   176712                  ;ERROR REGISTER
                                RPDS:   176710                  ;DRIVE STATUS REGISTER
                                RPCA1:  176723
                                VECTOR=254                      ;INTERRUPT VECTOR ADDR.
                                STATUS=256                      ;DISK INTERRUPT STATUS

                                ;MESSAGES

003100' 005015  047125  052111  MES1:   .ASCIZ  <15><12>/UNIT /
003106' 000040
003110' 005015  050122  051105  MES1A:  .ASCIZ  <15><12>/RPER= /
003116' 020075     000
003121'    015  051012  041520  MES2:   .ASCIZ  <15><12>/RPCS= /
003126' 036523  000040
003132' 005015  050122  051504  MES2A:  .ASCIZ  <15><12>/RPDS= /
003140' 020075     000
003143'    015  041412  046131  MES3:   .ASCIZ  <15><12>/CYLINDER= /
003150' 047111  042504  036522
003156' 000040
003160' 005015  051124  041501  MES4:   .ASCIZ  <15><12>/TRACK= /
003166' 036513  000011
003172' 005015  042523  052103  MES5:   .ASCIZ  <15><12>/SECTOR= /
003200' 051117  020075     000
003205'    015  042012  052101  MES12:  .ASCIZ  <15><12>/DATA COMPARE ERROR/
003212' 020101  047503  050115
003220' 051101  020105  051105
003226' 047522  000122
003232' 005015  052502  020123  MES13:  .ASCIZ  <15><12>/BUS ADDRESS INCORRECT/
003240' 042101  051104  051505
003246' 020123  047111  047503
003254' 051122  041505  000124
003262' 005015  054105  042520  MES18:  .ASCIZ  <15><12>/EXPECTED /
003270' 052103  042105  000040
003276' 005015  042522  042503  MES19:  .ASCIZ  <15><12>/RECEIVED /
003304' 053111  042105  000040
003312' 000000                  ENDP:   0                       ;START OF BUFFER AREA
        000001                          .END
```

DZRPGB.P11 CROSS REFERENCE TABLE -- USER SYMBOLS

```
ATTN    000540R          399     417*    483
ATTNB   000536R          398*    399*    404     413     416*    483*    484     493
BUFF    003050R          544     553     577     587     588     685*    705     815*
B0    = ******  GX       676     725     758
B1    = ******  GX       731
B10   = ******  GX       394     470     495
B11   = ******  GX       513
B14   = ******  GX       585
B15   = ******  GX       331     341     352     361     547     572     702     726
B3    = ******  GX       334     357     367
B9    = ******  GX       345
CLRTAB  000060R          319*    322
DEVTBL  002632R          319*    331*    341*    352     361     363     437*    445     447*    448*    462*    463*    464
                         465     473*    474*    486*    487     504     520*    522*    523*    554     555     563*    564*
                         566*    579     580     603*    608     609     707     752     756     790*    791
DSKER   002436R          704     723*
DSKINT  002336R          316     702*
DUMP  = ******  GX       512     717     720     742     745     763     766
ENDP    003312R          672     684     860*
ERROR = ******  GX       312
EXGEN   002070R          634     639*
EXPS    002626R          598*    768*
EXREF   002126R          651*    656
EXTINT  002444R          722     725*
EXTMEN  002076R          348     647*
FLAG    003054R          817*
HINUM = ******  GX       441     451
HLT   = ******  GX       389     396     408     412     472     490     497     501     506     549     574     602     695
                         712     723
HLTCT$= ******  GX       731     758
HOME    000404R          354     386*
INT     003052R          556*    581*    691     725*    816*
INTERR  003046R          558*    561     582*    696*    721*    724*    814*
JMPTBL  000550R          364     427*
LONUM = ******  GX       440     450     675
LOOP    000266R          359*    381
MAIN    000274R          361*    368
MAXREF  002154R          648     660*
MAXRF1  002162R          654     661*
MES1    003100R          734     833*
MES1A   003110R          741     835*
MES10   001262R          508     526*
MES11   001311R          511     530*
MES12   003205R          601     848*
MES13   003232R          714     852*
MES18   003262R          716     762     856*
MES19   003276R          719     765     858*
MES2    003121R          744     837*
MES2A   003132R          737     839*
MES3    003143R          747     841*
MES4    003160R          750     844*
MES5    003172R          755     846*
MEXIT   000372R          377*
MEXIT1  000402R          372     374     381*
MSG     002464R          731*
PASSCT  003034R          350*    369*    370     373     809*
```

DZRPQB.P11 CROSS REFERENCE TABLE -- USER SYMBOLS

```
PC     =.000007            354*  364*  377*  414*  475*  492*  521*  524*  542*  545*  560*  567*  584*
                           604*  639*  686*  697*  739*  760*  767*
PRINT  = ******  GX        507   510   600   713   715   718   733   736   740   743   746   749   754
                           761   764
PRINTR= ******  GX         739
PRI4   = ******  GX        557
PRI7   = ******  GX        313   315   647   649
PSW    = ******  GX        311*  55*   647*
RANADR  002214R            542   671*
RAND   = ******  GX        439   449   674
RANDAT  001732R            545   608*
RANDA1  001756R            612*  638
RAND1   002072R            608*  610   619   631*  641*
RAND2   002074R            609*  611   621   635*  642*
RDCNT   001724R            575   586   596   603*
READD   001530R            430   565   571*
RECS    002630R            503*  504   599*  769*
RPBA    003064R            553*  577*  578*  710   822*
RPCA    003066R            464*  554*  579*  823*
RPCA1   003076R            503   827*
RPCS    003056R            327*  390*  410   466*  499   515*  516   559*  583*  585   702   819*
RPCS1   003060R            328*  386*  438*  489*  494*  546*  571*  820*
RPDA    003070R            465*  555*  580*  824*
RPDS    003074R            329   387   394   404   413*  470   484   493*  495   513   518   547   572
                           726   738   826*
RPER    003072R            825*
RPWC    003062R            552*  576*  821*
R0     =%000000            318*  319*  320   321   325*  331*  332*  338*  341*  342*  400*  401*  403*
                           406   610*  614*  619*  623*  629*  631   632   689*  690*  693
R1     =%000001            326*  328   333*  334   337*  339   344*  345   375*  377   544*  587*  590*
                           591   599   632*  636*
R2     =%000002            588*  591   598   613*  616*  622*  625*  627*  628*  629
R3     =%000003            589*  593*  594   612*  617*  677*  680   682*  705*  706*  709*  710
R4     =%000004            363*  364   391*  392*  397*  399   467*  468*  482*  483   611*  615*  620*
                           621*  624*  626*  630*  635   636   671*  672*  673*  678
R5     =%000005            348*  351*  352   356*  360*  361   363   366*  437*  445   447*  448*  462*
                           463*  464   465   473*  474*  486*  487   504   520*  522*  523*  554   555
                           563*  564*  566*  579   580   603*  608   609   666*  707   752   756
SAVE    002212R            650*  651   652   655*  660*  665*  667*  671
SCOPES= ******  GX         314
SDUMP  = ******  GX        509   735   748   753   757
SEEK    000560R            427   437*
SEKCK   001040R            428   482*
SHIFT   001764R            614*  618
SP     =%000006            310*
START   000000R            310*
STATUS= 000256             317*  829*
STKPTR= ******  GX         310
SWR    = ******  GX        339   370*
TTY    = ******  GX        738*
UNIT    003032R            349*  355*  357   359*  365*  367   386   397   438   482   489   494   546
                           571   808*
UNIT1   002652R            792*  793
UNIT2   002672R            794*  795
UNIT3   002712R            796*  797
UNIT4   002732R            798*  799
```

```
UNIT5    002752R           800*    801
UNIT6    002772R           802*    803
UNIT7    003012R           804*    805
VECTOR=  000254            316*    828*
WAIT     002310R           560     584     689*
WORK     003036R           543*    633*    637*    810*
WORK1    003040R           440*    442*    443     445     447     450*    452*    453     455*    460*    462     675*    676*
                           678     680*    684*    685     811*
WORK2    003042R           441*    448     451*    456*    457     459*    460     812*
WORK3    003044R           751*    752*    756*    813*
WRITE    001354R           429     542*
WRTER    001522R           550     562     566*
.      = 003314R           517     519     791*    793*    795*    797*    799*    801*    803*    805*
```

DZRPGB.P11 CROSS REFERENCE TABLE -- PERMANENT SYMBOLS

```
ADC     620  622  624  625  627  630
ADD     332  342  356  366  578  590  619  621  623  626  628  629  655  684  706
        709
ASL     614
BEQ     324  335  346  358  368  372  446  496  505  514  634  664  703  708  711
        727
BGE     454  458
BHI     488  679
BIC     413  442  452  456  676  680
BICB    493
BIS     331  341  725
BIT     334  339  345  352  357  361  367  394  404  470  484  495  513  547  572
        585  702  726  731  758
BLT     444
BMI     330  388
BNE     322  340  353  362  374  393  395  402  405  407  469  471  485  548  562
        573  586  592  595  618  638  653  692  694  732  759
BPL     411  500  517  519
BR      336  381  491  498  502  565  654  656  683
CLC     343  681
CLR     318  319  325  326  338  349  350  351  359  360  398  400  463  520  556
        558  566  581  582  589  603  613  662  689  751
CMP     321  373  443  445  453  457  487  504  591  594  652  678  707  710
DEC     392  468  617  633  637
INC     333  355  365  369  401  403  473  474  486  522  523  563  564  593  690
        696  721  724
JMP     550  575  596  704  722
JSR     348  354  364  377  542  545  560  584  739
MOV     310  311  312  313  314  315  316  317  327  337  363  370  375  391  397
        437  440  441  447  448  450  451  462  465  467  482  543  544  552  553
        554  555  557  576  577  579  580  587  588  598  599  608  609  610  611
        612  631  632  635  636  647  648  649  650  661  671  675  677  685  705
        738
MOVB    328  386  390  399  438  460  464  466  483  489  494  503  515  546  559
        571  583  752  756
NOP     378  379  380
RESET   347  376
ROL     344  615  616
ROR     682
RTI     728
RTS     414  475  492  521  524  567  604  639  666  686  697  760  767
SUB     455  459  660  665  672  673
TST     320  323  329  371  387  406  410  499  518  561  651  663  691  693
TSTB    516
.ASCIZ  526  530  833  835  837  839  841  844  846  848  852  856  858
.BYTE   417
.END    861
.EVEN   421  536  789
.REM     19

. ABS.  000000    000
        003314    001

% ERRORS DETECTED:  12
```

```
.MAIN.  MACY11 27(732)  16-SEP-76  15:47  PAGE 25
DZRPGB.P11      CROSS REFERENCE TABLE -- PERMANENT SYMBOLS

% DEFAULT GLOBALS GENERATED:  25

 *,DZRPGB.SEQ/SOL/CRF/PAGNUM=DZRPGB
 RUN-TIME:  2 7 1 SECONDS
 RUN-TIME RATIO: 154/11=13.4
 CORE USED:  6K  (11 PAGES)
```

Spooler runtime 3 Seconds, 21 KCS, 86 disk reads, 3 disk writes, 23 pages