

RH11-RS03

BASIC FUNCTIONAL DIAL
MD-11-DZRSB-E

EP-DZRSB-E-DL-A

NOV 1976

COPYRIGHT © 1976

digital

FICHE 1 OF 1

MADE IN USA

This microfiche card contains a grid of frames. The first column on the left contains 12 frames, each with a header and a list of data. The remaining columns contain various types of data, including lists, tables, and diagrams. The frames are arranged in a grid that is 12 frames high and 10 frames wide. The data is printed in a small, dense font, typical of microfiche technology.


```

.REM      %

```

IDENTIFICATION

PRODUCT CODE:	MAINDEC-11-DZRSB-E-D
PRODUCT NAME:	RH11-RS03-RS03/LA-RS04 BASIC FUNCTION DIAGNOSTIC
DATE CREATED:	AUGUST 1976
MAINTAINER:	DIAGNOSTIC GROUP

THE INFORMATION IN THIS DOCUMENT IS SUBJECT TO CHANGE WITHOUT NOTICE AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT CORPORATION. DIGITAL EQUIPMENT CORPORATION ASSUMES NO RESPONSIBILITY FOR ANY ERRORS THAT MAY APPEAR IN THIS DOCUMENT.

THE SOFTWARE DESCRIBED IN THIS DOCUMENT IS FURNISHED UNDER A
LICENSE AND MAY ONLY BE USED OR COPIED IN ACCORDANCE WITH THE
TERMS OF SUCH LICENSE.

DIGITAL EQUIPMENT CORPORATION ASSUMES NO RESPONSIBILITY FOR THE
USE OR RELIABILITY OF ITS SOFTWARE ON EQUIPMENT THAT IS NOT
SUPPLIED BY DIGITAL.

COPYRIGHT (C) 1973, 1974, 1975, 1976 BY DIGITAL EQUIPMENT CORPORATION

[illegible]

C01

MAINDEC-11-DZRSB-E
DZRSBE.P11

RH11-RS03LA-RS03-RS04 BASIC FUNCTION DIAGNOSTIC MACY11 27(732) 25-SEP-76 09:06 PAGE 3

MAINDEC-11-DZRSB-E RH11-RS03/LA-RS04 BASIC FUNCTION DIAGNOSTIC PAGE 2
TABLE OF CONTENTS

CONTENTS

1. ABSTRACT
2. REQUIREMENTS
 - 2.1 EQUIPMENT
 - 2.3 PRELIMINARY PROGRAMS
3. LOADING PROCEDURE
4. STARTING PROCEDURE
 - 4.1 CONTROL SWITCH SETTINGS
 - 4.2 STARTING ADDRESS
 - 4.3 PROGRAM AND/OR OPERATING PROCEDURE
5. OPERATIONAL SWITCH SETTINGS
 - 5.2 SUBROUTINE ABSTRACT
6. ERRORS
7. RESTRICTIONS
8. MISCELLANEOUS
 - 8.1 EXECUTION TIME
 - 8.2 STACK POINTER
9. WRITE LOCK TEST
10. TEST DESCRIPTION

48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92

MAINDEC-11-DZRSB-E RH11-RS03/LA-RS04 BASIC FUNCTION DIAGNOSTIC PAGE 3
DESCRIPTION

1. ABSTRACT

THIS DIAGNOSTIC WAS DESIGNED TO TEST RS03,RS03/LA AND RS04 DRIVES.

THIS IS A BASIC FUNCTION DIAGNOSTIC WHICH IS USED TO VERIFY THAT THE (RH11) CONTROLLER AND THE (RS03,RS03/LA OR RS04) DISKS ARE OPERATING CORRECTLY. THIS IS NOT A RELIABILITY DIAGNOSTIC AND THEREFORE SHOULD NOT BE USED AS ONE. THIS PROGRAM CAN TEST UP TO 8 DRIVES. THE DRIVES CAN BE INTERMIXED AND IN ANY ORDER.

IF THE OPERATOR WOULD LIKE TO CHECK THE DISK REGISTERS PRIOR TO ENTERING THIS DIAGNOSTIC, THERE ARE SOME ROUTINES IN THE BACK OF THE DIAGNOSTIC WHICH CAN BE USED. THESE ROUTINES WILL ALLOW THE OPERATOR TO LOAD THE REGISTERS THROUGH THE SWITCHES. PLEASE REFERENCE THE STARTING ADDRESSES THAT WILL TEST THE REGISTERS YOU DESIRE.

2. REQUIREMENTS

2.1 EQUIPMENT

PDP11 STANDARD COMPUTER WITH A MINIMUM OK 8K OF MEMORY, AND AN RH11 CONTROLLER WITH A RS03, RS03/LA OR RS04 DISK.

2.3 PRELIMINARY PROGRAMS

NONE

3. LOADING PROCEDURE

USE STANDARD PROCEDURE FOR ABS TAPES.

4. STARTING PROCEDURE

4.1 CONTROL SWITCH SETTINGS

SEE 5.1.1 (ALL DOWN FOR WORST CASE TESTING)

4.2 STARTING ADDRESS

93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148

MAINDEC-11-DZRSB-E RH11-RS03/LA-RS04 BASIC FUNCTION DIAGNOSTIC PAGE 4
DESCRIPTION

4.3 PROGRAM AND/OR OPERATOR ACTION

LOAD PROGRAM INTO MEMORY USING ABS LOADER.

1. STARTING ADDRESS 200.

- A. SET SWITCHES (SEE SEC 5.1.1) ALL DOWN FOR WORST CASE. IF SWITCHLESS CPU, SIMPLY
- B. PRESS START.
- C. THE PROGRAM WILL LOOP AND BELL WILL RING ONCE EVERY PASS
- D. THE DISPLAY ON THE 11/45 WILL SHOW THE ITERATION COUNT IN THE LEFT BYTE AND TEST NUMBER IN THE RIGHT. TO USE, SET THE DATA DISPLAY SWITCH TO THE DISPLAY POSITION.
- E. THE PROGRAM WILL TEST ALL RS03, RS03/LA AND RS04 DISKS.

2. STARTING ADDRESSES FOR TESTING THE RH11-RS03/LA/04 REGISTERS USING THE SWITCH REGISTER. ON SWITCH-LESS MACHINES THESE ROUTINES ARE USEFULL FOR SCOPING. SIMPLY STRIKE ↑G ANYTIME AFTER PRESSING START TO ENTER OR CHANGE VALUE DESIRED.

A.	250	WORD COUNT REGISTER TEST
B.	254	BUS ADDRESS REG. TEST
C.	260	DISK ADDRESS REG. TEST
D.	264	DRIVE STATUS REG. TEST
E.	270	ERROR REG. TEST
F.	274	LOOK AHEAD REG. TEST
G.	300	RSCS2 REG. TEST
H.	304	ATTENTION SUMMARY REG. TEST
I.	310	MAINTENANCE REG. TEST
J.	314	RSCS1 REG TEST

5. OPERATIONAL SWITCH SETTINGS

THIS PROGRAM HAS BEEN MODIFIED TO RUN ON A PROCESSOR WITH OR WITHOUT A HARDWARE SWITCH REGISTER. WHEN FIRST EXECUTED THE PROGRAM TESTS THE EXISTENCE OF A HARDWARE SWITCH REGISTER. IF NOT FOUND A SOFTWARE SWITCH REGISTER LOCATION (SWREG=LOC.176) IS DEFAULTED TO. IF THIS IS THE CASE, UPON EXECUTION THE CONTENTS OF THE SWREG ARE DUMPED IN OCTAL ON THE CONSOLE TTY AND ANY CHANGES ARE REQUESTED

(I.E.) SWR=XXXXXX NEW=

POSSIBLE RESPONSES ARE:

- 1. <CR> IF NO CHANGES ARE TO BE MADE.
- 2. 6 DIGITS 0-7 TO REPRESENT IN OCTAL THE NEW SWITCH REGISTER VALUE; LAST DIGIT FOLLOWED BY <CR>.
- 3. ↑U TO ALLOW REENTERING VALUE IF ERROR IS COMMITTED

F01

MAINDEC-11-DZRSB-E
DZRSBE.P11

RH11-RS03LA-RS03-RS04 BASIC FUNCTION DIAGNOSTIC MACY11 27(732) 25-SEP-76 09:06 PAGE 6

205

KEYING IN SWREG VALUE.

MAINDEC-11-DZRSB-E RH11-RS03/LA-RS04 BASIC FUNCTION DIAGNOSTIC PAGE 5
DESCRIPTION

BUILT INTO THE PROGRAM IS THE ABILITY TO DYNAMICALLY CHANGE THE CONTENTS OF SWREG DURING PROGRAM EXECUTION. BY STRIKING +G (CNTL G) ON CONSOLE TTY THE OPERATOR SETS A REQUEST FLAG TO CHANGE THE CONTENTS OF SWREG, WHICH IS PROCESSED IN KEY AREAS OF THE PROGRAM CODE (IE) ERROR ROUTINES, AFTER HALTS END OF PASS, AND OTHER APPLICABLE AREAS.

SWITCH SETTINGS ARE:

```

SW<15> = 1 ..... HALT ON ERROR
SW<14> = 1 ..... LOOP ON TEST
SW<13> = 1 ..... INHIBIT TYPEOUTS
SW<12> = 1 ..... INHIBIT OBUFSV FROM CHANGING WHEN LOOKING
                FOR MEMORY ON -B- PORT
SW<11> = 1 ..... INHIBIT ITERATIONS OF SUBTEST
SW<10> = 1 ..... BELL ON ERROR
        0 ..... BELL ON PASS COMPLETE
SW<09> = 1 ..... LOOP ON ERROR
SW<08> = 1 ..... LOOP ON TEST IN SW<7:0>

```

5.2 SUBROUTINE ABSTRACTS

5.2.1 SCOPE

THIS SUBROUTINE CALL IS PLACED BETWEEN EACH SUBTEST IN THE INSTRUCTION SECTION. IT RECORDS THE STARTING ADDRESS OF EACH SUBTEST AS IT IS BEING ENTERED IN LOCATION "LAD". IF A SCOPE LOOP IS REQUESTED, THE CURRENT SUBTEST WILL BE LOOPED UPON. SW<11> ON A 1 INHIBITS ITERATION OF SUBTESTS. THE CONTENTS OF LAD MAY BE USED TO DETERMINE THE LAST SUBTEST SUCCESSFULLY COMPLETED.

5.2.2 HLT

THIS ROUTINE PRINTS OUT AN ERROR MESSAGE (SEE 6.1). TO INHIBIT
TYPEOUTS, PUT SW<13> ON A 1.

5.2.3 TRAPCATCHER

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

6. ERRORS

MAINDEC-11-DZRSB-E RH11-RS03/LA-RS04 BASIC FUNCTION DIAGNOSTIC PAGE 6
DESCRIPTION

6.1 ERROR PRINTOUT

THE FORMAT IS AS FOLLOWS:

ADR	CS1	=	-----	CS2	=	-----	ER	=	-----
GOOD		=	-----	BAD	=	-----			

WHERE:

CS1, CS2, ER ETC.	= RS11 DISK REGISTERS.
GOOD	= EXPECTED DATA.
BAD	= DATA RECEIVED.

TO FIND THE FAILING TEST, LOOK AT THE LISTING ABOVE THE ADDRESS TYPED.

6.2 ERROR RECOVERY

RESTART AT 200

7. RESTRICTIONS

NONE

8. MISCELLANEOUS

8.1 EXECUTION TIME

A BELL WILL RING WITHIN 1 MINUTE WITH ALL SWITCHES DOWN.

8.2 STACK POINTER

STACK IS INITALLY SET TO 500

9. WRITE LOCK TEST

THE WRITE LOCK TEST REQUIRES OPERATOR INTERVENTION. THE STARTING ADDRESS FOR THIS TEST IS 220. THE PROGRAM WILL TELL THE OPERATOR WHICH SWITCHES HAVE TO BE SET.

262
263
264
265
266
267
268
269
270
271
272
273
274
275
276
277
278
279
280
281
282
283
284
285
286
287
288
289
290
291
292
293
294
295
296
297
298
299
300
301
302
303
304
305
306
307
308
309
310
311
312
313
314
315
316
317

101

MAINDEC-11-DZRSB-E
DZRSBE.P11

RH11-RS03LA-RS03-RS04 BASIC FUNCTION DIAGNOSTIC MACY11 27(732) 25-SEP-76 09:06 PAGE 9

318
319

10. TEST DESCRIPTION

MAINDEC-11-DZRSB-E RH11-RS03/LA-RS04 BASIC FUNCTION DIAGNOSTIC PAGE 7
DESCRIPTION

1. TEST RSCS2

CLEAR ALL READ/WRITE BITS AND CHECK. SET ALL R/W BITS AND CHECK. NOW CLEAR AND RECHECK.

2. TEST FOR ONLINE DRIVES

SET ERROR BITS IN RSER. THIS CAUSES ATTENTION SUMMARY BITS TO SET IN RSAS. DO FOR ALL DRIVES. RSAS HAS NOT YET BEEN TESTED. SO IN THE CASE OF NO BITS IN RSAS SETTING, DRIVE 0 IS TESTED.

3. RESET TEST FOR REGISTERS

SET ALL R/W BITS IN RSCS1, RSCS2, RSBA, RSDA, RSER, RSWC, RSDB, AND RSMR. DO A RESET AND TEST ALL R/W BITS TO BE CLEARED.

4. SET AND CLEAR ALL REGISTERS

SET ALL R/W BITS IN RSCS1, RSCS2, RSBA, RSDA, RSER, RSWC, RSDB AND RSMR AND TEST. SET ALTERNATE BITS AND CHECK TO MAKE SURE BITS ARE NOT TIED TOGETHER. NOW SET ALL BITS AND CLEAR THEM TO MAKE SURE ALL CAN BE CLEARED ONCE SET.

5. RANDOM NUMBER TEST FOR RSWC AND RSDA

THIS TEST GENERATES RANDOM NUMBERS AND LOADS THEM INTO RSWC, RSDA AND RSBA.

6. TEST "CLEAR BIT" IN RSCS2

SET ALL R/W BITS IN RSCS1, RSCS2, RSBA, RSDA, RSER, RSWC, RSDB, AND RSMR. SET CLEAR BIT IN RSCS2. NOW TEST ALL R/W BITS FOR 0 IN ALL THE ABOVE REGISTERS.

7. TEST DLT AND TRE BITS

DO A READ FROM THE SILO. THIS SHOULD CAUSE A DLT AND A TRE ERROR BECAUSE THE SILO IS EMPTY.

8. CLEAR DLT AND TRE

CLEAR BY SETTING TRE IN RSCS1 AND TEST.

9. LOAD RSDB WITH ALL ONES AND ALL ZEROS

LOAD RSDB WITH A WORD OF ZEROS AND A WORD OF ONES. WAIT FOR "OR" TO SET AND THEN CHECK OUTPUT OF SILO. IF OR DID NOT SET ERROR MESSAGE APPEARS.

10. TEST FOR 66 LOCATIONS IN SILO

320
321
322
323
324
325
326
327
328
329
330
331
332
333
334
335
336
337
338
339
340
341
342
343
344
345
346
347
348
349
350
351
352
353
354
355
356
357
358
359
360
361
362
363
364
365
366
367
368
369
370
371
372
373
374
375

K01

MAINDEC-11-DZRSB-E
DZRSBE.P11

RH11-RS03LA-RS03-RS04 BASIC FUNCTION DIAGNOSTIC MACY11 27(732) 25-SEP-76 09:06 PAGE 11

376
377

THIS IS DONE BY PUTTING A BINARY COUNT IN EVERY LOCATION AND

MAINDEC-11-DZRSB-E RH11-RS03/LA-RS04 BASIC FUNCTION DIAGNOSTIC PAGE 8
DESCRIPTION

CHECKING THE OUTPUT FOR 66 WORDS.

11. TEST DLT ERROR

THIS IS DONE BY LOADING THE SILO WITH 67 WORDS WITHOUT READING ANY OUT. THIS SHOULD CAUSE DLT TO SET.

12. FLOAT A "1" AND A "0" THROUGH THE SILO

LOAD THE SILO WITH A WORD OF ZEROS AND FLOAT A "1" THROUGH THE WORD. THEN LOAD THE SILO WITH A WORD OF ALL ONES AND FLOAT A "0" THROUGH THE WORD. CHECK THE OUTPUT OF THE SILO FOR THE CORRECT ANSWER.

13. TEST NO-OP FUNCTION

THE NO-OP FUNCTION IS TESTED WITH AND WITHOUT ERROR BITS SET. ALL THE REGISTERS ARE CHECKED AFTER BOTH CASES.

14. TEST DRIVE CLEAR FUNCTION

FIRST SET ALL R/W BITS IN RSDA, RSWC, RSER, AND RSMR. DO A DRIVE CLEAR FUNCTION. NOW TEST ALL REGISTERS FOR CORRECT DATA.

15. EXECUTE A ONE WORD WRITE FUNCTION

SET RSWC TO -1. MOV -1 INTO OUTBUF. LOAD RSBA WITH OUTBUF. DO A WRITE TEST RDY BIT FOR 0 THEN WAIT FOR IT TO SET. TIME OUT TO ERROR IF RDY BIT DOESN'T SET AND CHECK FOR ERROR CONDITIONS. TEST RSDA FOR CORRECT ADDRESS. TEST WORD COUNT FOR 0. THIS IS TESTED ON -A- AND -B- PORT.

16. EXECUTE A ONE WORD WRITE CHECK

SET UP RSDA, RSBA, RSWC AND OUTBUF AS IN THE WRITE FUNCTION TEST. DO A WRITE CHECK FUNCTION. TEST RDY AS DONE IN THE WRITE TEST. CHECK FOR WRITE CHECK ERROR. THEN TEST RSDA, RSWC AND RSBA FOR CORRECT DATA. THIS IS TESTED ON -A- AND -B- PORT.

17. TEST READ FUNCTION

SETUP RSDA, RSBA, RSWC AND OUTBUF AS IN THE WRITE FUNCTION DO A READ FUNCTION. TEST RDY BIT AS DONE IN THE WRITE FUNCTION. TEST FOR ERRORS. ALSO TEST RSDA, RSWC AND RSBA FOR CORRECT DATA. THIS IS TESTED ON -A- AND -B- PORT.

18. TEST BLOCK SEARCH FUNCTION, PIP AND DRY BITS AND ADDR. CONF. BIT

DO A BLOCK SEARCH FOR SECTOR 32, LOOP ON ADDR. CONF. BIT IN

378
379
380
381
382
383
384
385
386
387
388
389
390
391
392
393
394
395
396
397
398
399
400
401
402
403
404
405
406
407
408
409
410
411
412
413
414
415
416
417
418
419
420
421
422
423
424
425
426
427
428
429
430
431
432
433

MO1

MAINDEC-11-DZRSB-E
DZRSBE.P11

RH11-R503LA-R503-R504 BASIC FUNCTION DIAGNOSTIC MACY11 27(732) 25-SEP-76 09:06 PAGE 13

434
435

RSMR. IF IT DOESN'T SET, TIMEOUT. WHEN YOU GET THERE DO A
BLOCK SEARCH FOR SECTOR 0. NOW WE KNOW THAT WE HAVE TIME TO

MAINDEC-11-DZRSB-E RH11-RS03/LA-RS04 BASIC FUNCTION DIAGNOSTIC PAGE 9
DESCRIPTION

TEST FOR DRY AND PIP BITS BEFORE FINDING SECTOR 0. FOR PIP SHOULD SET AND DRY SHOULD CLEAR BEFORE FINDING SECTOR 0. ONCE SECTOR 0 IS FOUND PIP SHOULD CLEAR AND DRY SHOULD SET. IF DRY DOES NOT SET A TIME OUT ERROR WILL OCCUR INDICATING SECTOR 0 WAS NOT FOUND. SC IN RSCS1 SHOULD ALSO SET. RSBA AND RSWC SHOULD NOT MOVE, THIS IS ALSO TESTED.

19. ILLEGAL FUNCTION CODE TEST

IN THIS TEST RSBA, RSWC AND RSDA ARE SET UP AS IF TO DO A LEGAL FUNCTION. AN ILLEGAL FUNCTION IS THEN EXECUTED. THE PROGRAM TEST FOR ILF AND ERR BITS TO SET. RSBA, RSWC AND RSDA ARE ALSO TESTED FOR CORRECT DATA. THIS IS DONE FOR ALL THE ILLEGAL FUNCTIONS.

FOR AN AID IN TROUBLE SHOOTING THE ILLEGAL FUNCTION CODE CAN BE LOADED INTO LOCATION ILLTAB OR ILFTB2, DEPENDING ON WHICH ILLEGAL FUNCTION TEST YOU WISH TO LOOP ON. IN THE NEXT LOCATION, FOLLOWING THE ILLEGAL FUNCTION, A 0 MUST BE LOADED. NOW BY SETTING SWITCH 14 (LOOP ON TEST), YOU WILL LOOP ON THE ILLEGAL FUNCTION.

20. TEST PAR IN RSER

SET PAR IN RSER AND CHECK. ALSO TEST ERR IN RSDS TO SET BECAUSE OF THE PAR SETTING.

21. TEST DPR AND MOL IN RSDS

BOTH THESE BITS SHOULD BE SET IN RSDS IF THE DRIVE IS ON LINE AND UP TO SPEED.

22. LOOK AHEAD TEST

FIRST CHECK TO SEE IF SECTOR FRACTION BITS ARE MOVING. NOW SET RSDA TO 0 AND INCREMENT IT EVERY TIME THE ADDR.CONF BIT SETS. IF THE ADDR.CONF BIT DOES NOT SET IN A CERTAIN LENGTH OF TIME, A TIME OUT ERROR OCCURS.

23. PARITY TEST

THIS WILL TEST THE PARITY LOGIC ONLY IF THERE IS PARITY MEMORY ON THE SYSTEM IN LESS THAN 28K. IT WILL WRITE BAD PARITY IN A MEMORY LOCATION THEN TRY TO DO A WRITE TO THE DRIVE FROM THAT LOCATION. THIS SHOULD CAUSE A PARITY ERROR.

24. TEST WRITE CHECK ERROR

IN THIS TEST THE PROGRAM WRITES A -1 ON TO THE DISK. A 0 IS NOW FLOATED THROUGH THE WORD IN THE BUS ADDRESS LOCATION, AND A WRITE CHECK FUNCTION IS DONE. THE WCE BIT IN RSCS2 SHOULD SET AND SHOULD CAUSE THE TRE BIT IN RSCS1 TO SET. THESE BITS

B02

MAINDEC-11-DZRSB-E
DZRSBE.P11

RH11-RS03LA-RS03-RS04 BASIC FUNCTION DIAGNOSTIC MACY11 27(732) 25-SEP-76 09:06 PAGE 15

492
493

ARE THEN CLEARED. A WORD OF 0 IS NOW WRITTEN ON THE DISK AND
A 1 IS FLOATED THROUGH THE WORD IN THE BUS ADDRESS AND THE

WRITE CHECK FUNCTION TEST IS REPEATED.

25. TEST PROGRAM ERROR BIT IN RSCS2

HERE THE PROGRAM ATTEMPTS TO INITIATE A DATA TRANSFER OPERATION WHILE THE CONTROL IS CURRENTLY PERFORMING ONE. THIS SHOULD CAUSE PGE TO SET WHICH SHOULD CAUSE TRE TO SET. THESE BITS ARE THEN CLEARED. RSWC IS ALSO TESTED FOR IT SHOULD NOT BE 0 FOR THE CURRENT OPERATION SHOULD HAVE BEEN ABORTED DUE TO THE PGE ERROR.

26. TEST RMR IN RSR

HERE A WRITE COMMAND IS GIVEN AND DURING ITS EXECUTION THE PROGRAM TRYS TO MODIFY THE RSDA REG. THIS SHOULD CAUSE THE RMR BIT TO SET WHICH CAUSES THE ERR BIT TO SET. THESE BITS ARE THEN CLEARED.

27. TEST DCK IN RSR

HERE A WRITE COMMAND IS GIVEN THEN DURING THIS FUNCTION A DRIVE CLEAR COMMAND IS GIVEN. THIS SHOULD CAUSE THE DCK BIT TO SET WHICH SHOULD CAUSE THE ERR BIT TO SET. THESE BITS ARE THEN CLEARED.

28. TEST DISK ADDRESS REGISTER

LOAD THE LAST DISK ADDRESS (7777) INTO RSDA. DO A ONE WORD WRITE AND CHECK THAT RSDA INCREMENTED TO 10000.

29. TEST IAE ERROR

DO A ONE WORD WRITE BUT FIRST SET RSDA TO AN INVALID ADDRESS SUCH AS 10000. THIS SHOULD CAUSE A IAE ERROR WHICH WILL CAUSE ERR, ATA AND SC BITS TO SET. THESE BITS ARE THEN CLEARED BY LOADING A 1 INTO ATA IN RSAS.

30. TEST FOR NON-EXISTENT DISK ERROR

FIRST FIND A DRIVE THAT IS NOT ON THE SYSTEM OR OFF LINE. NOW TRY TO DO A ONE WORD WRITE TO THAT DRIVE. NED IN RSCS2 SHOULD SET WHICH SHOULD CAUSE TRE TO SET. THESE BITS ARE THEN CLEARED BY MOVING A 1 INTO TRE.

31. TEST DAO IN RSR AND LBT IN RSDS

SET RSDA TO ITS LAST ADDRESS. NOW WRITE ONE SECTOR PLUS ONE WORD. DAO SHOULD SET AND LBT SHOULD SET. THESE SHOULD CAUSE ERR, ATA, TRE AND SC TO SET. THESE ARE CLEARED BY DOING A CLEAR.

32. TEST BAI IN RSCS2

494
495
496
497
498
499
500
501
502
503
504
505
506
507
508
509
510
511
512
513
514
515
516
517
518
519
520
521
522
523
524
525
526
527
528
529
530
531
532
533
534
535
536
537
538
539
540
541
542
543
544
545
546
547
548
549

D02

MAINDEC-11-DZRSB-E
DZRSBE.P11

RH11-RSD3LA-RSD3-RSD4 BASIC FUNCTION DIAGNOSTIC MACY11 27(732) 25-SEP-76 09:06 PAGE 17

550
551

SET BAI IN RSCS2. DO A ONE WORD WRITE AND CHECK RSBA TO SEE

MAINDEC-11-DZRSB-E RH11-RS03/LA-RS04 BASIC FUNCTION DIAGNOSTIC PAGE 11
DESCRIPTION

IF IT INCREMENTED.

33. TEST NON-EXISTENT MEMORY ERROR BIT IN RSCS2

SET BITS A16 AND A17 IN RSCS1 FOR AN 18 BIT ADDRESS. MOV 173000 INTO RSBA. MOV -1000 INTO RSWC AND DO A WRITE FUNCTION. THE NEM BIT SHOULD SET AND SHOULD CAUSE TRE TO SET. CLEAR THESE BITS BY LOADING A 1 INTO TRE.

34. TEST FOR ZERO'S IN A PARTIALLY FILLED SECTOR

FIRST WRITE A COMPLETE SECTOR WITH ALL ONES. THEN DO A ONE WORD WRITE. THE REMAINING 63 WORDS SHOULD BE WRITTEN AS ZERO'S. NOW DO A WRITE CHECK TO COMPARE FOR THESE ZERO'S.

35. PRIORITY INTERRUPT TEST

HERE THE PROGRAM ENABLES THE INTERRUPT AND DOES A ONE WORD WRITE FUNCTION. THE PROGRAM SHOULD NOT TRAP UNTIL THE PROCESSOR IS DROPPED TO PRIORITY 4.

36. DYNAMIC FUNCTION TEST

WHILE ONE DRIVE IS READING, THE UNIT # IN RSCS2 IS MODIFIED. IF THERE IS ANOTHER DRIVE ON THE SYSTEM, A DRIVE SEARCH IS PERFORMED ON IT. THIS IS ALL DONE WHILE THE FIRST DRIVE IS STILL READING.

552
553
554
555
556
557
558
559
560
561
562
563
564
565
566
567
568
569
570
571
572
573
574
575
576
577
578
579
580
581
582


```

%
.TITLE MAINDEC-11-DZRSB-E      RH11-RS03LA-RS03-RS04 BASIC FUNCTION DIAGNOSTIC
;COPYRIGHT 1973,1974,1975,1976 DIGITAL EQUIPMENT CORP., MAYNARD, MASS.
;PROGRAM BY STANLEY HARACKIEWICZ

```

```

583
584
585
586
587
588
589
590
591      100000
592      040000
593      020000
594      010000
595      004000
596      002000
597
598      001000
599      000400
600      000000
601      000046
602      000046 022026
603      000052
604      000052 040000
605
606      000174
607      000174 000000
608      000176 000000
609
610      000200
611      000200 000137 000230
612
613      000220
614      000220 000137 024354
615
616      000230
617      000230 042737 004000 001160
618      000236 000137 001202
619
620
621
622
623      000250 000137 023466
624      000254 000137 023512
625      000260 000137 023536
626      000264 000137 023562
627      000270 000137 023606
628      000274 000137 023632
629      000300 000137 023642
630      000304 000137 023666
631      000310 000137 023712
632      000314 000137 023736
633
634
635      000320 000320 004000 001160
636      000326 000137 001202
637
638

```

```

;
; SWITCH
;-----
SW15= 100000
SW14= 40000
SW13= 20000
SW12= 10000
SW11= 4000
SW10= 2000
SW9= 1000
SW8= 400
.= 0
.= 46
SENDAD 52
BIT14
.= 174
DISPREG: 0
SWREG: 0
.= 200
JMP @#STRT
.= 220
JMP @#WRTLCK ;WRITE LOCK TEST
.= 230
STRT: BIC @#BIT11,ONCEE ;CLEAR TKSEL TEST FLAG
JMP @#BEGIN ;STARTING ADDRESS IS BEGIN
;STARTING ADDRESSES FOR SWITCH REGISTER TESTING OF RH-RS REGISTERS
.= 250
JMP @#SRSMC ;WORD COUNT REG
JMP @#SR5BA ;BUS ADDR REG
JMP @#SR5DA ;DISK ADDR REG
JMP @#SR5DS ;DRIVE STATUS REG
JMP @#SR5ER ;ERROR REG
JMP @#SR5LA ;LOOK AHEAD REG
JMP @#SR5S2 ;CS2 REG
JMP @#SR5AS ;ATTENTION SUMMARY REG
JMP @#SR5MR ;MAINTENANCE REG
JMP @#SR5CS1 ;CS1 REG
.= 320
BIS @#BIT11,ONCEE ;SET TKSEL TEST FLAG
JMP @#BEGIN ;ENABLES OPERATOR TO DO A
;WRITE OR READ TO A DESIRED ADDR
;OPTIONAL WC, DA, AND UNIT #

```


G02

MAINDEC-11-DZRSB-E
DZRSBE.P11

RH11-RS03LA-RS03-RS04 BASIC FUNCTION DIAGNOSTIC MACY11 27(732) 25-SEP-76 09:06 PAGE 20

639

640
641
642
643
644
645
646
647
648
649
650
651
652
653
654
655
656
657
658
659
660
661
662
663
664
665
666
667
668
669
670
671
672

000001
104000
177776
177776
000007
000000
000001
000002
000003
000004
000005
000006
000007
000001
000002
000004
000010
000020
000040
000100
000200
000400
001000
002000
004000
010000
020000
040000
100000
000001
000000

N= 1
HLT= EMT
PS= 177776
PSW= PS
BELL= 7
R0= %0
R1= %1
R2= %2
R3= %3
R4= %4
R5= %5
SP= %6
PC= %7
BIT0= 1
BIT1= 2
BIT2= 4
BIT3= 10
BIT4= 20
BIT5= 40
BIT6= 100
BIT7= 200
BIT8= 400
BIT9= 1000
BIT10= 2000
BIT11= 4000
BIT12= 10000
BIT13= 20000
BIT14= 40000
BIT15= 100000
GOOD= %1
BAD= %0

;INITALIZE FOR NEWTST
;SET HLT TO EMT FOR ERROR TYPEOUTS
;PROCESSOR STATUS
;PROCESSOR STATUS WORD
;BELL
;R0 - DEFINE REGISTERS
;R1
;R2
;R3
;R4
;R5
;R6 - STACK POINTER
;R7 - PROGRAM COUNTER
;BIT EQUATES

;FOR GOOD DATA
;FOR BAD DATA


```

673      001000
674
675      001000 000000
676      001002 000000
677      001004 000000 000000
678      001010 000000
679      001012 000000
680      001014 001000
681      001016 177564
682      001020 177560
683      001022 177562
684      001024 177566
685      001026 177570
686      001030 177570
687
688      001100 001100
689      001100 000000
690      001102 000000
691
692
693
694      001104 172040
695      001106 172050
696      001110 172042
697      001112 172044
698      001114 172046
699      001116 172052
700      001120 172054
701      001122 172056
702      001124 172060
703      001126 172062
704      001130 172064
705      001132 172066
706      001134 000204
707      001136 000206
708      001140 172041
709      001142 172051
710      001144 172043
711      001146 172045
712
713
714
715      177572
716      172340
717      172342
718      172344
719      172356
720      172300
721      172302
722      172304
723      172316
724      000006
725      000000
726

```

```

.=      1000
ICNT:    0
ERRORS:  0
PCNT:    0,0
LAD:      0
HLTADR:  0
FILCHR:  1000
TPS:      177564
TKS:      177560
TKB:      177562
TPB:      177566
SWR:      177570
DISPLAY: 177570

```

```

.=      1100
SAVBAD:  0
OBUFSV:  0

```

;DISK I/O REGISTERS

```

RSCS1:    172040
RSCS2:    172050
RSMC:      172042
RSBA:      172044
RSDA:      172046
RSDS:      172052
RSEB:      172054
RSAS:      172056
RSLA:      172060
RSDB:      172062
RSMR:      172064
RSDT:      172066
RSVEC:     204
RSVCPS:    206
RSCS1B:    172041
RSCS2B:    172051
RSMCB:     172043
RSBAB:     172045

```

;MEMORY MANAGEMENT REGISTER ASSIGNMENTS

```

SR0=177572
KIPAR0=172340
KIPAR1=172342
KIPAR2=172344
KIPAR7=172356
KIPDR0=172300
KIPDR1=172302
KIPDR2=172304
KIPDR7=172316
RW=6
UP=00

```

```

;LH = ITERATION COUNT ;RH = TEST NO.
;ERROR COUNT
;2 WORD PASS COUNT
;LOOP ADDRESS FOR SCOPE
;ADDRESS OF LAST HLT INSTRUCTION EXECUTED
;FILCHR=0 (CHAR) ;FILCHR+1=2 (COUNT)
;OUTPUT STATUS REGISTER

```

```

;OUTPUT BUFFER
;SWITCH REGISTER
;DISPLAY REGISTER

```

```

;LOC FOR ILLEGAL FUNCTION CODE
;LOC OF OUTBUF

```

```

;DISK CONTROL + STATUS REGISTER
;DISK CONTROL + STATUS REGISTER
;WORD COUNT REGISTER
;BUS ADDRESS
;DISK ADDRESS (DESIRED ADDRESS)
;DRIVE STATUS
;ERROR REG.
;ATTENTION SUMMARY
;LOOK AHEAD
;DATA BUFFER REGISTER
;MAINTENANCE REGISTER
;DRIVE TYPE REGISTER
;INTERUPT VECTOR
;INTERUPT PRIO. VECTOR
;ODD BYTE ADD FOR CS1
;ODD BYTE ADD FOR CS2
;ODD BYTE ADD FOR CW
;ODD BYTE ADD FOR BA

```


727
728
729
730
731
732
733
734
735
736
737
738
739
740
741
742
743
744
745
746
747
748
749
750
751
752
753
754
755
756
757
758
759
760
761
762
763
764
765
766
767
768

000001
000002
000004
000010
000020
000040
000100
000200
000204
000210
000220
000240

040000
100000
000100
000200
002000
010000
040000
100000
000200
020000
002000
040000
100000
001000
100000
000010
000100

;BIT ASSIGNMENTS FOR ERROR TYPEOUTS
;THE RS REGISTERS ARE DIVIDED INTO 3 GROUPS.
;CS1,CS2 AND ER ARE IN THE FIRST GROUP.THIS GROUP IS ALWAYS
;TYPED WITH EITHER OF THE OTHER GROUPS. AS,BA,DA, WC AND DS
;ARE IN THE SECOND GROUP. DT,DB,MR, AND LA ARE IN THE 3RD
;GROUP.YOU CAN NOT INTERMIX GROUP 2 OR 3. THEY HAVE
;TO BE TYPED SEPERATELY.
;EXAMPLE: HLT !CS1 AS BA
; HLT !CS1!DT!DB

CS1=1	;CONTROL AND STATUS 1
ER=2	;CONTROL AND STATUS 2
DA=4	;DESIRED ADD
WC=10	;WORD COUNT
BA=20	;BUS ADDRESS
DS=40	;DRIVE STATUS
AS=100	;ATTENTION SUMMARY
CS2=200	;CONTROL AND STATUS REG
LA=204	;LOOK AHEAD
DB=210	;DATA BUFFER
MR=220	;MAINTENANCE
DT=240	;DRIVE TYPE

;BIT ASSIGNMENTS FOR THE REGISTER BITS

TRE=40000	;TRANSFER ERROR CS1
SC=100000	;SPECIAL CONDITIONS CS1
IR=100	;INPUT READY CS2
OR=200	;OUTPUT READY CS2
PGE=2000	;PROGRAM ERROR-CS2
NED=10000	;NON-EXISTENT DRIVE CS2
WCE=40000	;WRITE CHECK ERROR-CS2
DLT=100000	;DATA LATE ERROR CS2
DRY=200	;DRIVE READY DS
PIP=20000	;POSITIONING IN PROGRESS DS
LBT=2000	;LAST BLOCK TRANSFER-DS
ERR=40000	;ERROR DS
ATA=100000	;ATTENTION ACTIVE-DS
DAO=1000	;DISK OVERFLOW ERROR-ER
DCK=100000	;DATA CHECK ERROR-ER
BAI=10	;BUS ADDR INCREMENT INHIBIT
IE=100	;INTERRUPT INABLE CS1


```

769                                     ;WORKING LOCATIONS
770
771 001150 146723      RANNU: 146723      ;RANDOM NUMBER PRIME
772 001152 000000      UNNUM: 0           ;UNIT CURRENTLY BEING TESTED
773 001154 000000      UNITSV: 0          ;SET BIT=UNIT ON BUS
774 001156 000000      UNCMP: 0           ;FOR COMPARING FOR # OF DEVICE
775 001160 000000      ONCEE: 0           ;DID WE TEST ANY DRIVES
776 001162 000000      RS04DT: 0          ;CLR IF RS03 SET IF RS04
777 001164 000000      TIMSV: 0           ;SAVE LOC FOR TIME
778 001166 000000      AOB1: 0            ;PORT SWITCH
779                                     ;WRITE WRONG PARITY
780                                     ;PARITY REG
781 001170 000000      BPORTT: 0          ;BUFFER ADDR FOR -B- PORT
782 001172 000000      SAVEE: 0           ;WORK LOC
783
784                                     ;DISCRIPTION OF ONCEE BITS
785                                     ;BIT0 MEANS FOUND DRIVE
786                                     ;BIT11 DO TKSEL TEST
787                                     ;BIT12 TYPE COULD NOT FIND NED ONLY ONCE
788                                     ;BIT13 TYPE NO MEM ON B PORT ONLY ONCE
789                                     ;BIT14 0- DO WCE WITH 0 -1 DO WCE WITH 1
790                                     ;BIT15 MEANS ERROR FOUND
791
792                                     ;RH11 WORK REGISTERS
793                                     ;((CAN BE CHANGED IN ANY ROUTINE))
794 001174 000000      WORK: 0
795 001176 000000      WORK1: 0
796 001200 000000      WORK2: 0
797
798 001202 012706 000500      BEGIN: MOV #500,SP      ;SET STACK TO *** 500 ***
799 001206 012737 022770 000024      MOV #.POWER,2#24 ;SET UP PF VECTOR
800 001214 012737 000340 000026      MOV #340,2#26   ;LOCK OUT THE WORLD
801 001222 012737 022416 000030      MOV #.HLT,2#30   ;SET EMT VECTOR
802 001230 012737 000340 000032      MOV #340,2#32   ;LOCK UP
803 001236 012737 023422 000034      MOV #.TRAP,2#34  ;SET TRAP VECTOR
804 001244 012737 000340 000036      MOV #340,2#36   ;LOCK UP
805 001252 005037 001000      CLR ICNT              ;INIT ICNT
806 001256 005037 001010      CLR LAD               ;INIT LAD
807 001262 042737 143777 001160      BIC #143777,ONCEE ;CLEAR ONCEE
808 001270 012737 027364 001102      MOV #OUTBUF,OBUSV ;SAVE LOC OF OUTBUFFER
809 001276 032737 004000 001160      BIT #BIT11,ONCEE ;DO TKSEL TEST?
810 001304 001402                                     ;NO
811 001306 000137 024014      BEQ +6               ;YES
812 001312 104420      JMP 2#TKSEL                 ;SIZE FOR HDWR SWR
      SUSWR

```



```

813          :NOW TEST FOR DRIVES
814          :*****
815          :TEST 1          TEST FOR DRIVES
816          :*****
817 001314 104400      TST1: SCOPE
818
819 001316 012701 000010 MULTII: MOV      #8, R1          ;PUT 8 INTO R1 FOR COUNT
820 001322 005077 177560      CLR      @RSCS2        ;SET DEVICE TO ZERO
821 001326 012777 000007 177564 TRY: MOV      #7, @RSEB ;CAUSE AN ERROR +SETS BIT IN RSAS REG
822 001334 005301      DEC      R1                ;DO A MAXIMUM OF 8 TIMES
823 001336 001403      BEQ      DVNUM              ;TESTED FOR ALL DRIVES GET OUT
824 001340 005277 177542      INC      @RSCS2        ;INCREMENT DRIVE UNIT
825 001344 000770      BR       TRY                ;REPEAT FOR NEXT DRIVE
826 001346 017737 177550 001154 DVNUM: MOV      @RSAS, UNITSV ;SAVE
827 001354 012737 000401 001156      MOV      #401, UNCMP    ;SETUP TO CMP WITH UNITSV
828 001362 012737 000000 001152      MOV      #0, UNNUM      ;PUT 0 INTO UNIT NO.
829 001370 032777 020000 177430      BIT      @BIT13, @SWR   ;INHIBIT TYPE OUT?
830 001376 001015      BNE      STTEST             ;YES
831 001400 104402 001404      TYPE              ;ASCIZ <15><12>"TESTING UNIT "
832 001424 042737 100000 001160      BIC      @BIT15, ONCEE ;CLEAR ERROR FLAG
833 001432 033737 001156 001154 STTEST: BIT      UNCMP, UNITSV ;IS THIS DRIVE ON THE SYSTEM
834 001440 001475      BEQ      TRYNX              ;NO
835 001442 013777 001152 177436      MOV      UNNUM, @RSCS2   ;YES PUT UNIT # INTO CS2
836 001450 022777 000004 177454      CMP      #4, @RSDT      ;IS THIS A RS03LA?
837 001456 001004      BNE      7$                ;NO
838 001460 012737 000004 001162      MOV      #4, RS04DT     ;SETUP DRIVE TYPE FOR A LA DISK
839 001466 000435      BR       1$                ;GET OUT
840 001470 022777 000000 177434 7$: CMP      #0, @RSDT      ;IS THIS A RS03?
841 001476 001003      BNE      2$                ;NO
842 001500 005037 001162      CLR      RS04DT         ;YES
843 001504 000426      BR       1$                ;GET OUT
844 001506 022777 000001 177416 2$: CMP      #1, @RSDT      ;IS THIS A RS03 4US?
845 001514 001003      BNE      3$                ;NO
846 001516 005037 001162      CLR      RS04DT         ;RS03
847 001522 000417      BR       1$                ;GET OUT
848 001524 022777 000002 177400 3$: CMP      #2, @RSDT      ;IS THIS A RS04?
849 001532 001410      BEQ      6$                ;YES
850 001534 022777 000003 177370      CMP      #3, @RSDT      ;IS IT A RS04?
851 001542 001404      BEQ      6$                ;YES
852 001544 043737 001156 001154      BIC      UNCMP, UNITSV   ;CLEAR UNWANTED ATA BIT
853 001552 000430      BR       TRYNX              ;GET A NEW NUMBER
854 001554 052737 177777 001162 6$: BIS      #-1, RS04DT     ;YES RS04
855 001562 032777 020000 177236 1$: BIT      @BIT13, @SWR   ;INHIBIT TYPE OUT?
856 001570 001020      BNE      4$                ;YES
857 001572 032737 100000 001160      BIT      @BIT15, ONCEE ;ANY ERRORS?
858 001600 001404      BEQ      5$                ;NO
859 001602 104402 001606      TYPE              ;ASCIZ <15><12><12>
860 001612      5$:
861 001612 013746 001152      MOV      UNNUM, -(6)          ;PUT UNNUM ON STACK
862 001616 104406      TYPES                     ;TYPE STACK IN OCTAL - SUPPRESS
863 001620 104402 000040      TYPE              ;TYPE SPACE
864 001624 042737 100000 001160      BIC      @BIT15, ONCEE ;CLEAR ERROR FLAG
865 001632 000473      BR       NOWGO              ;NOW TEST

```



```

866 001634 006337 001156      TRYNX:  ASL      UNCMP      ;CHECK NEXT BIT FOR DRIVE
867 001640 103403              BCS      CHCKDV     ;DID WE TEST ANY REG?
868 001642 005237 001152      INC      UNNUM      ;INC UNIT #
869 001646 000671              BR       STTEST     ;CHECK FOR NEXT DRIVE
870
871 001650 032737 000001 001160 CHCKDV: BIT      #BIT0,ONCEE ;DID WE TEST ANY DRIVES?
872 001656 001057              BNE      DONEE      ;YES WE DID TEST A DRIVE
873 001660 012737 100000 001156      MOV      #100000,UNCMP ;NO DRIVES TESTED, COULD NOT SET
874 001666 005037 001152      CLR      UNNUM      ;ANY AS BITS, THUS DEFAULTS TO
875 001672 032777 020000 177126      BIT      #BIT13,ASWR ;INHIBIT TYPE OUT?
876 001700 001045              BNE      4$         ;YES
877 001702 013746 001152      MOV      UNNUM,-(6) ;PUT UNNUM ON STACK
878 001706 104406              TYPES             ;TYPE STACK IN OCTAL - SUPRESS
879 001710 104402 000040      TYPE      ,40        ;TYPE SPACE
880 001714 104402 001720      TYPE      ,.+2       ;ASCIZ <15><12>"COULD NOT FIND DRIVE WILL TEST DRIVE 0
881 002012 000000              HALT              ;WAIT
882 002014 000402              4$:  BR       NOWGO    ;TEST DRIVE 0
883 002016 000137 021762      DONEE: JMP      DONE     ;GET OUT
884
885 ;THIS TEST IS DESIGNED TO TEST THE ABILITY OF RESET
886 ;TO CLEAR ALL THE RH AND RS REGISTERS
887
888 002022 012737 027364 001102 NOWGO: MOV      #OUTBUF,OBUSV ;SAVE LOC OF OUTBUFFER
889 002030 052737 000001 001160      BIS      #BIT0,ONCEE ;SET FOUND DRIVE FLAG
890 002036 013737 022414 001164      MOV      TIMES,TIMSV ;SAVE TIME
891 002044 012737 000001 022414      MOV      #1,TIMES ;ONLY TEST ONCE
892 *****
893 ;TEST 2 RESET TEST FOR REGISTERS
894 *****
895 TST2: SCOPE
896 002052 104400              MOV      #340,3#PS ;LOCK OUT INTERRUPTS
897 002054 012737 000340 177776      MOV      UNNUM,3RSCS2 ;GET UNIT #
898 002062 013777 001152 177016      MOV      #177776,3RSCS1 ;SET ALL
899 002070 012777 177776 177006      MOV      #177777,3RSBA ;POSSIBLE R/W
900 002076 012777 177777 177006      MOV      #177777,3RSDA ;BITS IN THESE REGISTERS
901 002104 012777 177777 177002      MOV      #177777,3RSER
902 002112 012777 177777 177000      MOV      #177777,3RSMR
903 002120 012777 177777 177002      MOV      #177777,3RSWC
904 002126 012777 177777 176754      MOV      #177777,3RSCS2
905 002134 012777 177737 176744      MOV      #177737,3RSCS2
906 002142 000005              RESET ;CLEAR ALL BITS IN ALL REG.
907
908 ;TEST RSCS2 FOR CLEARED BITS
909 002144 022777 000100 176734      CMP      #100,3RSCS2 ;DID THESE BITS GET CLEARED?
910 002152 001401              BEQ      +4         ;YES
911 002154 104200              HLT      !CS2 ;(417) SHOULD BE CLEARED IN CS2
912 002156 013777 001152 176722      MOV      UNNUM,3RSCS2 ;PUT # OF UNIT IN TEST IN CS2
913 002164 022777 010600 176724      CMP      #10600,3RSDS ;IS DPR AND MOL SET?
914 002172 001401              BEQ      +4         ;YES
915 002174 104040              HLT      !DS ;NO WHY NOT?
916
917 ;TEST CONTROL AND STATUS REG 1
918 002176 022777 004200 176700      CMP      #4200,3RSCS1 ;DID THE READY BIT SET?
919 002204 001401              BEQ      +4         ;YES
920 002206 104001              HLT      !CS1 ;READY SHOULD BE SET

```



```

920                                     ;TEST BUS ADDRESS REGISTER
921
922 002210 005777 176676               TST   @RSBA           ;IS BA REG. CLEARED
923 002214 001401                     BEQ   :+4             ;YES
924 002216 104020                     HLT   !BA             ;SHOULD BE 0
925
926                                     ;TEST DISK ADDRESS REGISTER
927
928 002220 005777 176670               TST   @RSDA           ;IS DA CLEARED
929 002224 001401                     BEQ   :+4             ;YES
930 002226 104004                     HLT   !DA             ;SHOULD BE 0
931
932                                     ;TEST ERROR REG RSER
933
934 002230 005777 176664               TST   @RSER           ;DID RSER CLEAR?
935 002234 001401                     BEQ   :+4             ;YES
936 002236 104002                     HLT   !ER             ;BITS(157015) SHOULD BE CLEARED
937
938                                     ;TEST RS MAINTENANCE REGISTER
939
940 002240 032777 000077 176662       BIT   #77,@RSMR       ;DID THESE BITS GET CLEARED
941 002246 001401                     BEQ   :+4             ;YES
942 002250 104220                     HLT   !MR             ;BITS(77) SHOULD BE 0
943
944                                     ;TEST WC REG IT SHOULD NOT CHANGE
945
946 002252 022777 177777 176630       CMP   #177777,@RSWC   ;DID IT CHANGE?
947 002260 001401                     BEQ   :+4             ;NO
948 002262 104010                     HLT   !WC             ;RESET SHOULD NOT MODIFY RSWC
949
950                                     ;TEST RSAS
951
952 002264 005777 176632               TST   @RSAS           ;IS REG CLEAR
953 002270 001401                     BEQ   :+4             ;YES
954 002272 104100                     HLT   !AS             ;NO

```



```

955 ;*****
956 ;TEST 3 TEST CLEAR BIT IN CS2 ON ALL THE R/W BITS
957 ;*****
958 002274 104400 TST3: SCOPE
959
960 002276 012737 000340 177776 TTAGG: MOV #340, @PS ;LOCK OUT INTERRUPTS
961 002304 013777 001152 176574 MOV UNNUM, @RSCS2 ;GET UNIT #
962 002312 012777 043576 176564 MOV #43576, @RSCS1 ;SET ALL
963 002320 012777 177777 176564 MOV #177777, @RSBA ;POSSIBLE
964 002326 012777 177777 176560 MOV #177777, @RSDA ;REGISTERS
965 002334 012777 177777 176556 MOV #177777, @RSER
966 002342 012777 177777 176556 MOV #177777, @RSDB
967 002350 012777 177777 176532 MOV #177777, @RSWC
968 002356 012777 020417 176522 MOV #20417, @RSCS2
969 002364 012777 000071 176536 MOV #71, @RSMR
970 002372 012777 000040 176506 MOV #40, @RSCS2 ;CLEAR ALL BITS
971 002400 022777 000100 176500 CMP #100, @RSCS2 ;DID THE RIGHT BITS CLEAR?
972 002406 001401 BEQ +4 ;YES
973 002410 104200 HLT ;CS2 ;(417) SHOULD BE CLEARED IN CS2
974 002412 013777 001152 176466 MOV UNNUM, @RSCS2 ;GET DRIVE NUMBER
975 002420 032777 173577 176456 BIT #173577, @RSCS1 ;DID ALL BITS GET CLEARED
976 002426 001401 BEQ +4 ;YES
977 002430 104001 HLT ;CS1 ;NO, ALL BITS SHOULD BE 0
978 ;TEST BUS ADDRESS REGISTER
979
980 002432 005777 176454 TST @RSBA ;IS BA REG. CLEARED
981 002436 001401 BEQ +4 ;YES
982 002440 104020 HLT ;BA ;SHOULD BE 0
983
984 ;TEST DISK ADDRESS REGISTER
985
986 002442 005777 176446 TST @RSDA ;IS DA CLEARED
987 002446 001401 BEQ +4 ;YES
988 002450 104020 HLT ;BA ;SHOULD BE 0
989
990 ;TEST ERROR REG RSER
991
992 002452 005777 176442 TST @RSER ;DID THESE BITS GET CLEARED
993 002456 001401 BEQ +4 ;YES
994 002460 104002 HLT ;ER ;BITS(157015) SHOULD BE CLEARED
995
996 ;TEST RS MAINTENANCE REGISTER
997 002462 002777 000077 176440 BIT #77, @RSMR ;DID THESE BITS GET CLEARED
998 002470 001401 BEQ +4 ;YES
999 002472 104220 HLT ;MR ;BITS(77) SHOULD BE 0
1000
1001 ;TEST WC REG. IT SHOULD NOT CHANGE
1002 002474 022777 177777 176406 CMP #177777, @RSWC ;DID WC CHANGE
1003 002502 001401 BEQ +4 ;NO
1004 002504 104010 HLT ;WC ;WHY DID IT CHANGE?

```



```

1005 ;*****
1006 ;TEST 4          SET AND CLEAR ALL REGISTERS
1007 ;*****
1008 002506 104400 TST4: SCOPE
1009 ;CAN WE SET THE FUNCTION BITS IN THE RSCS1 REG.
1010 ;BITS 7,6,5,4,3,2&1
1011
1012 002510 104414 CLRDK          ;CLEAR ALL RS REG
1013 002512 013737 001164 022414 MOV      TIMSV,TIMES ;GET TIME
1014 002520 012777 003576 176356 MOV      #3576,RSCS1 ;SET DISK FUNCTION BITS
1015 002526 022777 007776 176350 CMP      #7776,RSCS1 ;ARE THESE BITS SET?
1016 002534 001401 BEQ      +4          ;NO
1017 002536 104001 HLT      !CS1        ;SHOULD = 7776
1018 002540 012777 002524 176336 MOV      #2524,RSCS1 ;SET THESE BITS
1019 002546 022777 006724 176330 CMP      #6724,RSCS1 ;DID THEY SET
1020 002554 001401 BEQ      +4          ;YES
1021 002556 104001 HLT      !CS1        ;SHOULD BE 6724
1022 002560 012777 001052 176316 MOV      #1052,RSCS1 ;SET THESE BITS
1023 002566 022777 005252 176310 CMP      #5252,RSCS1 ;ARE THEY =?
1024 002574 001401 BEQ      +4          ;YES
1025 002576 104001 HLT      !CS1        ;SHOULD = 5252
1026 002600 104400 TST5: SCOPE
1027 ;CLEAR THE FUNCTION BITS
1028
1029 002602 012777 043576 176274 MOV      #43576,RSCS1 ;SET DISK FUNCTION BITS
1030 002610 005077 176270 CLR      RSCS1
1031 002614 022777 004200 176262 CMP      #4200,RSCS1 ;IS THE READY BIT SET
1032 002622 001401 BEQ      +4          ;YES
1033 002624 104001 HLT      !CS1        ;RSCS1 SHOULD = 4200
1034
1035 ;*****
1036 ;TEST 6          TEST RSCS2
1037 ;*****
1038 002626 104400 TST6: SCOPE
1039
1040 002630 000005 RESET          ;CLEAR WORLD
1041 002632 022777 000100 176246 CMP      #100,RSCS2 ;DID THEY CLEAR?
1042 002640 001401 BEQ      +4          ;YES
1043 002642 104200 HLT      !CS2        ;NO
1044 002644 012777 021037 176234 MOV      #21037,RSCS2 ;SET BITS 21017
1045 002652 022777 021137 176226 CMP      #21137,RSCS2 ;DID THESE BITS GET SET
1046 002660 001405 BEQ      1$          ;YES
1047 002662 017700 176220 MOV      RSCS2,BAD
1048 002666 012701 021137 MOV      #21137,GOOD ;WHAT CS2 SHOULD =
1049 002672 104000 HLT                      ;CS2 = BAD GOOD = CORRECT ANS

```


003

MAINDEC-11-DZRSB-E
DZRSBE.P11

TST6

RH11-RS03LA-RS03-RS04 BASIC FUNCTION DIAGNOSTIC MACY11 27(732) 25-SEP-76 09:06 PAGE 30
TEST RSCS2

1050	002674	012777	020025	176204	15:	MOV	#20025,RSRCS2	:SET THESE BITS
1051	002702	022777	020125	176176		CMP	#20125,RSRCS2	:DID THESE BITS GET SET
1052	002710	001401				BEQ	+4	:YES
1053	002712	104200				HLT	:CS2	:NO,CS2 SHOULD = 20125
1054	002714	012777	000012	176164		MOV	#12,RSRCS2	:LOAD THESE BITS
1055	002722	022777	000112	176156		CMP	#112,RSRCS2	:DID THESE BITS GET SET IN CS2
1056	002730	001401				BEQ	+4	:YES
1057	002732	104200				HLT	:CS2	:BAD = CS2 GOOD = CORRECT ANS
1058	002734	012777	177777	176144		MOV	#-1,RSRCS2	:SET BITS
1059	002742	005077	176140			CLR	RSRCS2	:CLEAR THEM
1060	002746	022777	000100	176132		CMP	#100,RSRCS2	:DID CLEAR WORK
1061	002754	001401				BEQ	+4	:YES
1062	002756	104200				HLT	:CS2	:R/W BITS DID NOT CLEAR
1063	002760	013777	001152	176120		MOV	UNNUM,RSRCS2	:GET UNIT #
1064	002766	104400						
1065					TST7:	SCOPE		
1066					;CAN WE	SET ALL	THE RSBA BITS	
1067	002770	012777	177777	176114		MOV	#177777,RSRBA	:SET THE BITS
1068	002776	022777	177776	176106		CMP	#177776,RSRBA	:DID THEY SET
1069	003004	001401				BEQ	+4	:YES
1070	003006	104020				HLT	:BA	:BITS 17776 SHOULD BE SET
1071	003010	012777	125252	176074		MOV	#125252,RSRBA	:SET THESE BITS
1072	003016	022777	125252	176066		CMP	#125252,RSRBA	:ARE THEY =
1073	003024	001401				BEQ	+4	:YES
1074	003026	104020				HLT	:BA	:SHOULD BE 125252
1075	003030	012777	052524	176054		MOV	#52524,RSRBA	:SET THESE BITS
1076	003036	022777	052524	176046		CMP	#52524,RSRBA	:ARE THEY =
1077	003044	001401				BEQ	+4	:YES
1078	003046	104020				HLT	:BA	:SHOULD BE 52524
1079								
1080	003050	104400			TST10:	SCOPE		
1081					;FLOAT A 1	THROUGH RSBA		
1082								
1083	003052	012701	000002		FLOTBA:	MOV	#2,GOOD	:GET A 2
1084	003056	000241				CLC		:CLEAR CARRY
1085	003060	010177	176026		15:	MOV	GOOD,RSRBA	:FLOAT NUMBER
1086	003064	017700	176022			MOV	RSRBA,BAD	:GET BA
1087	003070	020100				CMP	GOOD,BAD	:COMPARE BA
1088	003072	001401				BEQ	+4	:BA CORRECT
1089	003074	104000				HLT		:BAD=BA GOOD=CORRECT ANS
1090	003076	006101				ROL	GOOD	:ROTATE NUMBER
1091	003100	103367				BCC	15	:LOOP TILL DONE

E03

MAINDEC-11-DZRSB-E
DZRSBE.P11 TST6RH11-RS03LA-RS03-RS04 BASIC FUNCTION DIAGNOSTIC MACY11 27(732) 25-SEP-76 09:06 PAGE 31
TEST RSC52

```

1092 003102 104400          TST11: SCOPE
1093
1094          ;CLEAR THE RSBA REGISTER
1095
1096 003104 012777 177777 176000      MOV      #177777, @RSBA      ;SET RSBA EQUAL TO ALL ONES
1097 003112 005077 175774              CLR      @RSBA
1098 003116 005777 175770              TST      @RSBA      ;TEST FOR BIT0 SET IN RSBA (READ ONLY BIT)
1099 003122 001401              BEQ      .+4      ;YES
1100 003124 104020              HLT      !BA      ;NO
1101 003126 104400          TST12: SCOPE
1102
1103          ;CAN WE SET ALL BITS IN RSWC REGISTER
1104
1105 003130 012777 177777 175752      MOV      #177777, @RSWC      ;SET WC BITS
1106 003136 022777 177777 175744      CMP      #177777, @RSWC      ;ARE ALL BITS SET
1107 003144 001401              BEQ      .+4      ;YES
1108 003146 104010              HLT      !WC      ;NO
1109 003150 012777 125252 175732      MOV      #125252, @RSWC      ;SET THESE BITS
1110 003156 022777 125252 175724      CMP      #125252, @RSWC      ;ARE THEY =
1111 003164 001401              BEQ      .+4      ;YES
1112 003166 104010              HLT      !WC      ;SHOULD BE 125252
1113 003170 012777 052525 175712      MOV      #52525, @RSWC      ;SET THESE BITS
1114 003176 022777 052525 175704      CMP      #52525, @RSWC      ;ARE THEY =
1115 003204 001401              BEQ      .+4      ;YES
1116 003206 104010              HLT      !WC      ;SHOULD BE 152525
1117 003210 104400          TST13: SCOPE
1118
1119          ;FLOAT A 1 THROUGH RSWC
1120
1121 003212 012701 000001      FLOTWC: MOV      #1, GOOD      ;GET A 1
1122 003216 000241              CLC              ;CLEAR CARRY
1123 003220 010177 175664      1$:  MOV      GOOD, @RSWC      ;FLOAT NUMBER
1124 003224 017700 175660      MOV      @RSWC, BAD      ;GET WC
1125 003230 020100              CMP      GOOD, BAD      ;COMPARE WC
1126 003232 001401              BEQ      .+4      ;WC CORRECT
1127 003234 104000              HLT              ;BAD=WC GOOD=CORRECT ANS
1128 003236 006101              ROL              ;ROTATE NUMBER
1129 003240 103367              BCC      1$      ;LOOP TILL DONE

```



```

1130          ;CLEAR THE WORD COUNT REGISTER
1131 003242 104400 TST14: SCOPE
1132
1133 003244 012777 177777 175636 MOV #177777, @RSCW ;SET RSCW REGISTER EQUAL TO ALL ONES
1134 003252 005077 175632 CLR @RSCW
1135 003256 005777 175626 TST @RSCW ;DID ALL BITS GET CLEARED
1136 003262 001401 BEQ .+4 ;YES
1137 003264 104010 HLT !WC ;NO
1138 003266 104400 TST15: SCOPE
1139
1140          ;CAN WE SET ALL THE BITS IN THE RSDA REGISTER.
1141
1142 003270 012777 177777 175616 MOV #177777, @RSDA ;SET ALL BITS
1143 003276 022777 177777 175610 CMP #177777, @RSDA ;ARE THE BITS SET
1144 003304 001401 BEQ .+4 ;YES
1145 003306 104004 HLT !DA ;NO
1146 003310 012777 125252 175576 MOV #125252, @RSDA ;SET THESE BITS
1147 003316 022777 125252 175570 CMP #125252, @RSDA ;ARE THEY =
1148 003324 001401 BEQ .+4 ;YES
1149 003326 104004 HLT !DA ;SHOULD BE 125252
1150 003330 012777 052525 175556 MOV #52525, @RSDA ;SET THESE BITS
1151 003336 022777 052525 175550 CMP #52525, @RSDA ;ARE THEY =
1152 003344 001401 BEQ .+4 ;YES
1153 003346 104004 HLT !DA ;SHOULD BE 52525
1154 003350 104400 TST16: SCOPE
1155
1156          ;FLOAT A 1 THROUGH RSDA
1157
1158 003352 012701 000001 FLOTDA: MOV #1, GOOD ;GET A 1
1159 003356 000241 CLC ;CLEAR CARRY
1160 003360 010177 175530 15: MOV GOOD, @RSDA ;FLOAT NUMBER
1161 003364 017700 175524 MOV @RSDA, BAD ;GET DA
1162 003370 020100 CMP GOOD, BAD ;COMPARE DA
1163 003372 001401 BEQ .+4 ;DA CORRECT
1164 003374 104000 HLT ;BAD=DA GOOD=CORRECT ANS
1165 003376 006101 ROL GOOD ;ROTATE NUMBER
1166 003400 103367 BCC 15 ;LOOP TILL DONE

```



```

1167                                     ;CAN WE CLEAR THE RSDA REG.
1168 003402 104400 TST17: SCOPE
1169
1170 003404 012777 177777 175502      MOV      #177777, @RSDA      ;SET RSDA TO ALL ONES
1171 003412 005077 175476              CLR      @RSDA
1172 003416 005777 175472              TST      @RSDA      ;TEST FOR ZERO RSDA
1173 003422 001401              BEQ      +4      ;YES
1174 003424 104004              HLT      !DA      ;ANS SHOULD BE 0
1175 003426 104400 TST20: SCOPE
1176
1177                                     ;SET AND CLEAR THE RSER REG.
1178
1179 003430 012777 177777 175462      MOV      #177777, @RSER      ;SET THESE BITS
1180 003436 022777 177017 175454      CMP      #177017, @RSER      ;DID THEY SET
1181 003444 001401              BEQ      +4      ;YES
1182 003446 104002              HLT      !ER      ;RSER SHOULD = 157017
1183 003450 112777 000001 175442      MOVB     #1, @RSER      ;A MOVB INST
1184 003456 022777 000001 175434      CMP      #1, @RSER      ;SHOULD MODIFY COMPLETE WD
1185 003464 001401              BEQ      +4      ;OK
1186 003466 104002              HLT      !ER
1187
1188 003470 104400 TST21: SCOPE
1189
1190 003472 012777 052005 175420      MOV      #52005, @RSER      ;SET THESE BITS
1191 003500 022777 052005 175412      CMP      #52005, @RSER      ;DID THEY SET
1192 003506 001401              BEQ      +4      ;YES
1193 003510 104002              HLT      !ER      ;ER SHOULD = 52005
1194 003512 104400 TST22: SCOPE
1195
1196 003514 012777 125012 175376      MOV      #125012, @RSER      ;SET THESE BITS
1197 003522 022777 125012 175370      CMP      #125012, @RSER      ;DID THEY SET
1198 003530 001401              BEQ      +4      ;YES
1199 003532 104002              HLT      !ER      ;ER SHOULD = 105012

```


H03

MAINDEC-11-DZRSB-E
DZRSBE.P11 TST6RH11-R503LA-R503-R504 BASIC FUNCTION DIAGNOSTIC MACY11 27(732) 25-SEP-76 09:06 PAGE 34
TEST RSCS2

```

1200 003534 104400          TST23: SCOPE
1201
1202 003536 012777 177017 175354      MOV      #177017, @R5R      ;SET THESE BITS
1203 003544 005077 175350              CLR      @R5R          ;CLEAR THEM
1204 003550 005777 175344              TST      @R5R          ;DID THEY CLEAR
1205 003554 001401              BEQ      +4              ;YES
1206 003556 104002              HLT      !ER              ;SHOULD = 0
1207 003560 104400
1208
1209          TST24: SCOPE
1210          ;SET AND CLEAR RSMR
1211 003562 012777 000070 175340      MOV      #70, @RSMR      ;SET THESE BITS
1212 003570 017737 175334 001174      MOV      @RSMR, WORK    ;PUT INTO WORKABLE REG
1213 003576 042737 177700 001174      BIC      #177700, WORK  ;CLEAR JUNK
1214 003604 022737 000070 001174      CMP      #70, WORK    ;DID THEY SET
1215 003612 001401              BEQ      +4              ;YES
1216 003614 104220              HLT      !MR              ;SHOULD = 70
1217 003616 104400
1218          TST25: SCOPE
1219 003620 012777 000070 175302      MOV      #70, @RSMR      ;SET BITS
1220 003626 005077 175276              CLR      @RSMR          ;CLEAR THEM
1221 003632 032777 000077 175270      BIT      #77, @RSMR    ;DID THEY CLEAR
1222 003640 001401              BEQ      +4              ;YES
1223 003642 104220              HLT      !MR              ;BITS (77) SHOULD = 0
1224 003644 104400
1225          TST26: SCOPE
1226 003646 012777 000050 175254      MOV      #50, @RSMR      ;SET BITS
1227 003654 017737 175250 001174      MOV      @RSMR, WORK    ;PUT IN WORKABLE REG
1228 003662 042737 177700 001174      BIC      #177700, WORK  ;CLEAR JUNK
1229 003670 022737 000050 001174      CMP      #50, WORK    ;DID THESE BITS SET
1230 003676 001401              BEQ      +4              ;YES
1231 003700 104220              HLT      !MR              ;BITS (50) SHOULD BE SET
1232 003702 104400
1233          TST27: SCOPE
1234 003704 012777 000020 175216      MOV      #20, @RSMR      ;SET BITS
1235 003712 017737 175212 001174      MOV      @RSMR, WORK    ;PUT INTO WORKABLE REG
1236 003720 042737 177700 001174      BIC      #177700, WORK  ;CLEAR JUNK
1237 003726 022737 000020 001174      CMP      #20, WORK    ;DID THEY SET
1238 003734 001401              BEQ      +4              ;YES
1239 003736 104220              HLT      !MR              ;MR SHOULD AT LEAST HAVE A (21)

```



```

1240 ;*****
1241 ;TEST 30 LOAD RANDOM NUMBERS INTO RSWC, RSDA AND RSBA
1242 ;*****
1243 TST30: SCOPE
1244
1245 003740 104400 RANTS: MOV #1000,WORK ;MAKE TABLE 1000 WDS LONG
1246 003750 012701 027364 MOV #OUTBUF,R1 ;GET STARTING LOC OF TABLE
1247 003754 004537 026634 JSR R5,RANDOM ;GENERATE #
1248 003760 012704 027364 MOV #OUTBUF,R4 ;SETUP FOR COMPARE
1249 003764 012737 003772 001010 MOV #LOP1,LAD ;SETUP LOOP ADDR
1250 003772 012703 001000 LOP1: MOV #1000,R3 ;LOAD TEST COUNTER
1251 003776 005303 4S: DEC R3 ;DONE WITH COMPARE?
1252 004000 001413 BEQ 1S ;YES
1253 004002 013705 001110 MOV RSWC,R5 ;GET WC ADDRESS
1254 004006 011415 MOV (R4),(R5) ;LOAD WC
1255 004010 021524 CMP (R5),(R4)+ ;IS IT CORRECT?
1256 004012 001771 BEQ 4S ;YES
1257 004014 017700 175070 MOV @RSWC,BAD ;GET BAD WC
1258 004020 014401 MOV -(R4),GOOD ;GET GOOD ANS
1259 004022 104000 HLT ;TYPE THEM OUT
1260 004024 005724 TST (R4)+ ;UPDATE RANDOM NUMBER
1261 004026 000763 BR 4S ;CONT
1262 004030 012704 027364 1S: MOV #OUTBUF,R4 ;GET STARTING LOC OF TABLE
1263 004034 012737 004042 001010 MOV #LOP2,LAD ;SETUP LOOP ADDR
1264 004042 012703 001000 LOP2: MOV #1000,R3 ;SETUP TEST COUNTER
1265 004046 005303 3S: DEC R3 ;DONE YET?
1266 004050 001413 BEQ 1S ;YES
1267 004052 013705 001114 MOV RSDA,R5 ;LOAD DA ADDRESS INTO R5
1268 004056 011415 MOV (R4),(R5) ;LOAD DA
1269 004060 021524 CMP (R5),(R4)+ ;IS IT CORRECT?
1270 004062 001771 BEQ 3S ;YES
1271 004064 017700 175024 MOV @RSDA,BAD ;GET BAD DATA
1272 004070 014401 MOV -(R4),GOOD ;GET GOOD DATA
1273 004072 104000 HLT ;TYPE IT OUT
1274 004074 005724 TST (R4)+ ;UPDATE RANDOM NUMBER
1275 004076 000763 BR 3S ;CONTINUE
1276 004100 012704 027364 1S: MOV #OUTBUF,R4 ;GET STARTING LOC OF TABLE
1277 004104 012737 004112 001010 MOV #LOP3,LAD ;SETUP LOOP ADDR
1278 004112 012703 001000 LOP3: MOV #1000,R3 ;SETUP TEST COUNTER
1279 004116 005303 3S: DEC R3 ;DONE YET?
1280 004120 001416 BEQ 2S ;YES
1281 004122 013705 001112 MOV RSBA,R5 ;LOAD ADDRESS OF BA INTO R5
1282 004126 011415 MOV (R4),(R5) ;LOAD BA
1283 004130 042714 000001 BIC #BIT0,(R4) ;CLEAR BIT 0
1284 004134 021514 CMP (R5),(R4) ;IS IT CORRECT?
1285 004136 001767 BEQ 3S ;YES
1286 004140 017700 174746 MOV @RSBA,BAD ;GET BAD DATA
1287 004144 011401 MOV (R4),GOOD ;GET GOOD DATA
1288 004146 104000 HLT ;TYPE IT OUT
1289 004150 000400 BR 1S ;GET OUT
1290 004152 005724 1S: TST (R4)+ ;GET NEW NUMBER
1291 004154 000760 2S: BR 3S ;CONTINUE
1292 004156 000240 NOP

```



```

1293 ;*****
1294 ;TEST 31 TEST ODD BYTE INSTRUCTIONS ON CS1, CS2, WC AND BA
1295 ;*****
1296 004160 104400 TST31: SCOPE
1297
1298 004162 104414 BITST: CLRDK ;CLEAR ALL RS REG
1299 004164 012777 003566 174712 MOV #3566,ARSCS1 ;LOAD CS1
1300 004172 112777 000005 174740 MOVB #5,ARSCS1B ;LOAD BIT
1301 004200 022777 006766 174676 CMP #6766,ARSCS1 ;DID IT LOAD?
1302 004206 001401 BEQ +4 ;YES
1303 004210 104001 HLT !CS1
1304 004212 112777 000032 174664 MOVB #32,ARSCS1
1305 004220 022777 006632 174656 CMP #6632,ARSCS1
1306 004226 001401 BEQ +4
1307 004230 104001 HLT !CS1 ;CS1 SHOULD = 6632
1308
1309 004232 104400 TST32: SCOPE
1310
1311 004234 013777 001152 174644 BITCS2: MOV UNNUM,ARSCS2 ;LOAD UNIT NUMBER
1312 004242 052777 177400 174636 BIS #177400,ARSCS2 ;LOAD ALL BITS
1313 004250 105077 174666 CLRB ARSCS2B ;CLR UPPER BYTE
1314 004254 013701 001152 MOV UNNUM,GOOD ;GET UNIT NO.
1315 004260 052701 000100 BIS #100,GOOD ;SET OR BIT
1316 004264 017700 174616 MOV ARSCS2,BAD ;GET CS2
1317 004270 020001 CMP BAD,GOOD ;IS CS2 CORRECT?
1318 004272 001401 BEQ +4 ;YES
1319 004274 104000 HLT ;LOAD BYTE DID NOT WORK
1320
1321 004276 104400 TST33: SCOPE
1322
1323 004300 012777 025252 174602 BITWC: MOV #25252,ARSWC ;LOAD WC
1324 004306 112777 000377 174630 MOVB #377,ARSWCB ;LOAD BIT
1325 004314 022777 177652 174566 CMP #177652,ARSWC ;DID IT LOAD?
1326 004322 001401 BEQ +4 ;YES
1327 004324 104010 HLT !WC ;NO WC SHOULD =177652
1328 004326 112777 000123 174554 MOVB #123,ARSWC
1329 004334 022777 177523 174546 CMP #177523,ARSWC
1330 004342 001401 BEQ +4
1331 004344 104010 HLT !WC ;WC SHOULD = 177523
1332
1333 004346 104400 TST34: SCOPE
1334
1335 004350 012777 025252 174534 BITBA: MOV #25252,ARSBAB ;LOAD DA
1336 004356 112777 000377 174562 MOVB #377,ARSBAB ;LOAD BIT
1337 004364 022777 177652 174520 CMP #177652,ARSBAB ;DID IT LOAD?
1338 004372 001401 BEQ +4 ;YES
1339 004374 104020 HLT !BA ;DA SHOULD =177652
1340 004376 112777 000125 174506 MOVB #125,ARSBAB
1341 004404 022777 177524 174500 CMP #177524,ARSBAB
1342 004412 001401 BEQ +4
1343 004414 104020 HLT !BA ;BA SHOULD = 177525
1344 004416 104414 CLRDK ;CLEAR ALL RS REG

```


K03

MAINDEC-11-DZRSB-E
DZRSBE.P11 TST35RH11-RS03LA-RS03-RS04 BASIC FUNCTION DIAGNOSTIC MACY11 27(732) 25-SEP-76 09:06 PAGE 37
TEST DATA LATE IN CS2

```

1345 ;*****
1346 ;TEST 35 TEST DATA LATE IN CS2
1347 ;*****
1348 004420 104400 TST35: SCOPE
1349
1350 ;DO A READ FROM SILO: SHOULD GET DLT + TRE ERROR BECAUSE SILO IS EMPTY
1351
1352 SILOB: CLRDK ;CLEAR ALL RS REG
1353 004424 017700 174476 MOV JFSDB,BAD ;READ FROM EMPTY SILO
1354 004430 017700 174452 MOV JRSDB,BAD ;GET CS2
1355 004434 012701 100100 MOV #100100,GOOD ;GET CORRECT ANS
1356 004440 053701 001152 BIS UNNUM,GOOD ;FOR CS2
1357 004444 020001 CMP BAD,GOOD ;IS CS2 CORRECT?
1358 004446 001401 BEQ .+4 ;YES
1359 004450 104200 HLT !CS2 ;SHOULD HAVE DLT ERROR
1360 004452 022777 144200 174424 CMP #144200,JRSCS1 ;DID SC AND TRE SET?
1361 004460 001401 BEQ .+4 ;YES
1362 004462 104001 HLT !CS1 ;SC AND TRE SHOULD BE SET
1363 004464 012777 040000 174412 MOV #TRE,JRSCS1 ;CLEAR ERROR BIT
1364 004472 032777 140000 174404 BIT #140000,JRSCS1 ;DID SC + TRE CLEAR
1365 004500 001401 BEQ .+4 ;YES
1366 004502 104001 HLT !CS1 ;TRE AND SC SHOULD BE 0
1367 004504 017700 174376 MOV JRSDB,BAD ;GET CS2
1368 004510 042701 100000 BIC #BIT15,GOOD ;GET CORRECT ANS
1369 004514 020100 CMP GOOD,BAD ;IS CS2 CORRECT?
1370 004516 001401 BEQ .+4 ;YES
1371 004520 104200 HLT !CS2 ;DLT SHOULD BE 0
1372 ;*****
1373 ;TEST 36 LOAD RSDB WITH ALL ONES AND ALL ZEROS
1374 ;*****
1375 004522 104400 TST36: SCOPE
1376
1377 ZERONE: CLRDK ;CLEAR ALL RS REG
1378 004526 005077 174374 CLR JRSDB ;LOAD DB WITH ALL 0
1379 004532 012777 177777 174366 MOV #177777,JRSDB ;LOAD DB WITH ALL ONES
1380 004540 012737 002000 001174 MOV #2000,WORK ;TIME OUT ROUTINE
1381 004546 012701 000300 MOV #300,GOOD ;GET CORRECT FOR CS2
1382 004552 053701 001152 BIS UNNUM,GOOD
1383 004556 017700 174324 2$: MOV JRSDB,BAD ;GET CS2
1384 004562 020100 CMP GOOD,BAD ;IS IT CORRECT?
1385 004564 001404 BEQ 3$ ;YES
1386 004566 005337 001174 DEC WORK ;TO WAIT FOR OR
1387 004572 001371 BNE 2$ ;TO SET
1388 004574 104200 HLT !CS2 ;OR SHOULD BE SET
1389 004576 005001 3$: CLR GOOD ;LOAD BAD WITH DB
1390 004600 017700 174322 MOV JRSDB,BAD ;IS BAD CORRECT
1391 004604 020100 CMP GOOD,BAD ;YES
1392 004606 001401 BEQ .+4 ;COULD NOT FLOAT 0 THROUGH DB
1393 004610 104000 HLT ;LOAD GOOD WITH ANS
1394 004612 012701 177777 MOV #-1,GOOD ;GET DATA FROM DB
1395 004616 017700 174304 MOV JRSDB,BAD ;IS DB CORRECT
1396 004622 020100 CMP GOOD,BAD ;YES
1397 004624 001401 BEQ .+4 ;BAD SHOULD = 177777
1398 004626 104000 HLT

```


1399 004630 104400

TST37: SCOPE
;TEST FOR 66 LOCATIONS IN SILO PUT COUNT IN EVERY LOCATION

1400

1401

1402 004632 104414

1403 004634 005001

1404 004636 005201

1405 004640 010177 174262

1406 004644 022701 000102

1407 004650 001372

1408 004652 012701 000200

1409 004656 053701 001152

1410 004662 017700 174220

1411 004666 020100

1412 004670 001401

1413 004672 104200

1414 004674 005001

1415 004676 005201

1416 004700 022701 000103

1417 004704 001405

1418 004706 017700 174214

1419 004712 020100

1420 004714 001770

1421 004716 104000

1422 004720 032777 000200 174160

1423 004726 001401

1424 004730 104200

1425

1426

1427

1428 004732 005001

1429 004734 005201

1430 004736 010177 174164

1431 004742 022701 000103

1432 004746 001401

1433 004750 000771

1434 004752 032777 100000 174126

1435 004760 001001

1436 004762 104200

1437

1438

1439

1440 004764 017700 174136

1441 004770 012701 000001

1442 004774 020100

1443 004776 001401

1444 005000 104000

1445 005002 104400

SILO: CLRDK

1\$:

2\$:

3\$:

4\$:

;DOES SILO CHANGE WITH 67TH WORD: IT SHOULD NOT

TST40: SCOPE

CLRDK

CLR

INC

MOV

CMP

BNE

MOV

BIS

MOV

CMP

BEQ

HLT

CLR

INC

CMP

BEQ

MOV

CMP

BEQ

HLT

BIT

BEQ

HLT

CLR

INC

MOV

CMP

BEQ

BR

BIT

BNE

HLT

MOV

MOV

CMP

BEQ

HLT

R1

R1

R1,RSDB

#66.,R1

1\$

#200,GOOD

UNNUM,GOOD

RSDB,BAD

GOOD,BAD

+4

!CS2

GOOD

GOOD

#67.,GOOD

3\$

RSDB,BAD

GOOD,BAD

2\$

HLT

#OR,RSDB

+4

!CS2

R1

R1

R1,RSDB

#67.,R1

+4

4\$

#DLT,RSDB

+4

!CS2

RSDB,BAD

#1,GOOD

GOOD,BAD

+4

;CLEAR ALL RS REG

;CLEAR COUNTER

;INCREMENT COUNTER

;LOAD SILO

;LAST LOC. YET?

;NO LOOP AGAIN

;GET CORRECT ANS FOR CS2

;GET CS2

;IS CS2 CORRECT?

;YES

;OR SHOULD BE 1

;CLEAR LOCATION COUNTER

;ADD 1 TO IT

;LAST LOC YET?

;YES

;GET LOC FROM DB

;DO LOCATIONS MATCH?

;YES

;CAN NOT MATCH 66 LOCATIONS

;IS OR 0

;YES

;OR SHOULD BE 0

;CLEAR COUNTER

;ADD 1 TO COUNT

;PUT INTO COUNTER

;DONE YET?

;YES

;NO DO AGAIN

;DID DATA LATE SET?

;YES

;DLT DID NOT SET

;GET 1ST WD FORM SILO

;CORRECT ANS OF SILO

;IS SILO GOOD

;YES

;SILO SHOULD NOT HAVE MOVED

M03

MAINDEC-11-DZRSB-E
DZRSBE.P11 TST36

RH11-RS03LA-RS03-RS04 BASIC FUNCTION DIAGNOSTIC MACY11 27(732) 25-SEP-76 09:06 PAGE 39
LOAD RSDB WITH ALL ONES AND ALL ZEROS

```

1446 ;FLOAT A 1 AND A 0 THROUGH THE SILO
1447
1448 005004 104414 SILOFL: CLRDK ;CLEAR ALL RS REG
1449 005006 000241 CLC ;CLEAR CARRY TO FLOAT A 0
1450 005010 012701 000001 MOV #1,GOOD ;GET UP DATA FOR INPUT TO SILO
1451 005014 010177 174106 1$: MOV GOOD,RSDB ;LOAD DB
1452 005020 006101 ROL GOOD ;SHIFT BIT
1453 005022 103401 BCS .+4 ;DONE YET SHIFTING?
1454 005024 000773 BR 1$ ;NO
1455 005026 012701 177776 MOV #-2,GOOD ;SET ALL ONES
1456 005032 000261 SEC ;SET CARRY TO ROL
1457 005034 010177 174066 3$: MOV GOOD,RSDB ;LOAD SILO
1458 005040 006101 ROL GOOD ;SHIFT 0
1459 005042 103774 BCS 3$ ;LOOP TILL DONE
1460
1461 ;NOW TEST OUTPUT
1462
1463 005044 000241 CLC ;CLEAR CARRY
1464 005046 012701 000001 MOV #1,GOOD ;CORRECT ANS
1465 005052 017700 174050 2$: MOV RSDB,BAD ;GET DATA FROM DB
1466 005056 020100 CMP GOOD,BAD ;IS DB DATA GOOD?
1467 005060 001401 BEQ .+4 ;YES
1468 005062 104000 HLT ;DB COULD NOT BUBBLE CORRECTLY
1469 005064 006101 ROL GOOD ;SETUP FOR NEXT ANS
1470 005066 103401 BCS .+4 ;DONE YET?
1471 005070 000770 BR 2$ ;NO
1472 005072 012701 177776 MOV #-2,GOOD ;SETUP FOR ANS
1473 005076 017700 174024 4$: MOV RSDB,BAD ;GET DATA FROM DB
1474 005102 020100 CMP GOOD,BAD ;IS IT CORRECT?
1475 005104 001401 BEQ .+4 ;YES
1476 005106 104000 HLT ;DB WRONG
1477 005110 000261 SEC ;SET CARRY TO ROL
1478 005112 006101 ROL GOOD ;SETUP FOR NEXT ANS
1479 005114 103770 BCS 4$ ;LOOP TILL DONE

```



```

1480 ;*****
1481 ;TEST 41 TEST NO-OP FUNCTION
1482 ;*****
1483 005116 104400 TST41: SCOPE
1484
1485 NOOP: CLRDK
1486 005122 012737 177777 027364 MOV #177777,OUTBUF
1487 005130 013777 001102 173754 MOV @#OBUFSV,@RSBA
1488 005136 012777 177777 173744 MOV #-1,@RSWC
1489 005144 012777 000001 173732 MOV #1,@RSCS1
1490 005152 032777 000001 173724 BIT #1,@RSCS1
1491 005160 001401 BEQ .+4
1492 005162 104001 HLT !CS1
1493 005164 005777 173730 TST @RSER
1494 005170 001401 BEQ .+4
1495 005172 104002 HLT !ER
1496 005174 022777 177777 173706 CMP #-1,@RSWC
1497 005202 001401 BEQ .+4
1498 005204 104010 HLT !WC
1499 005206 005777 173702 TST @RSDA
1500 005212 001401 BEQ .+4
1501 005214 104004 HLT !DA
1502 005216 023777 001102 173666 CMP @#OBUFSV,@RSBA
1503 005224 001401 BEQ .+4
1504 005226 104020 HLT !BA
1505 005230 033777 001156 173664 BIT UNCMP,@RSAS
1506 005236 001401 BEQ .+4
1507 005240 104100 HLT !AS

```

```

;CLEAR ALL RS REG
;DATA TO BE XFERED
;SET UP CURRENT ADDRESS
;LOAD WC WITH -1
;DO NO-OP FUNCTION
;DID GO BIT CLEAR
;YES
;GO BIT SHOULD BE CLEARED
;DID ANY ERRORS OCCUR?
;NO
;ALL ERROR BITS SHOULD BE 0
;DID WC MOVE?
;NO
;WC SHOULD = 177777
;DID DA MOV
;NO
;DA SHOULD =0
;DID BA MOVE
;NO
;BA MOVED
;AS SHOULD NOT SET ON
;A NO-OP FUNCTION
;AS SET WHY?

```



```

1508 ;*****
1509 ;TEST 42 TEST NO-OP FUNCTION WITH ERROR BITS SET
1510 ;*****
1511 TST42: SCOPE
1512
1513 NN00P: CLRDK ;CLEAR ALL RS REG
1514 MOV #7,RSER ;LOAD ER
1515 BIT UNCMP,RSAS ;IS ATA BIT SET?
1516 BNE +4 ;YES
1517 HLT ;AS BIT SHOULD BE SET
1518 MOV #177777,OUTBUF ;DATA TO BE XFERED
1519 MOV #0BUFSV,RSBA ;SET UP CURRENT ADDRESS
1520 MOV #-1,RSWC ;LOAD WC WITH -1
1521 MOV #1,RSCSI ;DO NO-OP FUNCTION
1522 BIT #1,RSCSI ;DID GO BIT CLEAR
1523 BEQ +4 ;YES
1524 HLT ;CS1 ;GO BIT SHOULD BE CLEARED
1525 CMP #150600,RSOS ;DID ERR BITS SET?
1526 BEQ +4 ;NO
1527 HLT ;DS ;ERR BIT SHOULD BE 0
1528 CMP #-1,RSWC ;DID WC MOVE?
1529 BEQ +4 ;NO
1530 HLT ;WC ;WC SHOULD = 177777
1531 TST RSDA ;DID DA MOV
1532 BEQ +4 ;NO
1533 HLT ;DA ;DA SHOULD =0
1534 CMP #0BUFSV,RSBA ;DID BA MOVE
1535 BEQ +4 ;NO
1536 HLT ;BA ;BA MOVED
1537 BIT UNCMP,RSAS ;AS SHOULD BE SET
1538 BNE +4 ;IS IT?
1539 HLT ;AS ;NO
1540 CMP #7,RSER ;DID ER CHANGE?
1541 BEQ +4 ;NO
1542 HLT ;ER ;ER SHOULD NOT CHANGE

```


C04

MAINDEC-11-DZRSB-E
DZRSBE.P11 TST43RH11-R503LA-R503-R504 BASIC FUNCTION DIAGNOSTIC MACY11 27(732) 25-SEP-76 09:06 PAGE 42
TEST NO-OP FUNCTION CODE 21

```

1543 ;*****
1544 ;TEST 43 TEST NO-OP FUNCTION CODE 21
1545 ;*****
1546 005422 104400 TST43: SCOPE
1547
1548 005424 104414 NOOP21: CLRDK
1549 005426 012737 177777 027364 MOV #177777,OUTBUF ;CLEAR ALL RS REG
1550 005434 013777 001102 173450 MOV @#0BUFSV,@RSBA ;DATA TO BE XFERED
1551 005442 012777 177777 173440 MOV #-1,@RSWC ;SET UP CURRENT ADDRESS
1552 005450 012777 000021 173426 MOV #21,@RSCS1 ;LOAD WC WITH -1
1553 005456 032777 000001 173420 BIT #1,@RSCS1 ;DO NO-OP FUNCTION
1554 005464 001401 BEQ +4 ;DID GO BIT CLEAR
1555 005466 104001 HLT ;CS1 ;YES
1556 005470 005777 173424 TST @RSCR ;GO BIT SHOULD BE CLEARED
1557 005474 001401 BEQ +4 ;DID ANY ERRORS OCCUR?
1558 005476 104002 HLT ;ER ;NO
1559 005500 022777 177777 173402 CMP #-1,@RSWC ;ALL ERROR BITS SHOULD BE 0
1560 005506 001401 BEQ +4 ;DID WC MOVE?
1561 005510 104010 HLT ;WC ;NO
1562 005512 005777 173376 TST @RSDA ;WC SHOULD = 177777
1563 005516 001401 BEQ +4 ;DID DA MOV
1564 005520 104004 HLT ;DA ;NO
1565 005522 023777 001102 173362 CMP @#0BUFSV,@RSBA ;DA SHOULD =0
1566 005530 001401 BEQ +4 ;DID BA MOVE
1567 005532 104020 HLT ;BA ;NO
1568 005534 033777 001156 173360 BIT UNCMP,@RSAS ;BA MOVED
1569 005542 001401 BEQ +4 ;AS SHOULD NOT SET ON
1570 005544 104100 HLT ;AS ;A NO-OP FUNCTION
1571 005546 022777 004220 173330 CMP #4220,@RSCS1 ;AS SET WHY?
1572 005554 001401 BEQ +4 ;IS CS1 CORRECT?
1573 005556 104040 HLT ;DS ;YES
;CS1 SHOULD = 4220

```



```

*****
TEST 44          TEST NO-OP FUNCTION CODE 21 WITH ERROR BITS SET
*****
TST44: SCOPE

NNOP21: CLRDK
1574                                     ;CLEAR ALL RS REG
1575                                     ;LOAD ER
1576                                     ;IS ATA BIT SET?
1577 005560 104400                       ;YES
1578                                     ;AS BIT SHOULD BE SET
1579 005562 104414                       ;DATA TO BE XFERED
1580 005564 012777 000007 173326         ;SET UP CURRENT ADDRESS
1581 005572 033777 001156 173322         ;LOAD WC WITH -1
1582 005600 001001                       ;DO NO-OP FUNCTION
1583 005602 104100                       ;DID GO BIT CLEAR
1584 005604 012737 177777 027364         ;YES
1585 005612 013777 001102 173272         ;GO BIT SHOULD BE CLEARED
1586 005620 012777 177777 173262         ;DID ERR BITS SET?
1587 005626 012777 000021 173250         ;NO
1588 005634 032777 000001 173242         ;ERR BIT SHOULD BE 0
1589 005642 001401                       ;DID WC MOVE?
1590 005644 104001                       ;NO
1591 005646 022777 150600 173242         ;WC SHOULD = 1777777
1592 005654 001401                       ;DID DA MOV
1593 005656 104040                       ;NO
1594 005660 022777 177777 173222         ;DA SHOULD =0
1595 005666 001401                       ;DID BA MOVE
1596 005670 104010                       ;NO
1597 005672 005777 173216               ;BA MOVED
1598 005676 001401                       ;AS SHOULD BE SET
1599 005700 104004                       ;IS IT?
1600 005702 023777 001102 173202         ;NO
1601 005710 001401                       ;DID ER CHANGE?
1602 005712 104020                       ;NO
1603 005714 033777 001156 173200         ;ER SHOULD NOT CHANGE
1604 005722 001001                       ;IS CS1 CORRECT?
1605 005724 104100                       ;YES
1606 005726 022777 000007 173164         ;CS1 SHOULD = 104220
1607 005734 001401                       ;
1608 005736 104002                       ;
1609 005740 022777 104220 173136         ;
1610 005746 001401                       ;
1611 005750 104040                       ;

MOV      #7, ARSER
BIT      UNCMP, ARSAS
BNE      +4
HLT      !AS
MOV      #177777, OUTBUF
MOV      @#0BUFSV, ARSBA
MOV      #-1, ARSWC
MOV      #21, ARSCS1
BIT      #1, ARSCS1
BEQ      +4
HLT      !CS1
CMP      #150600, ARSDS
BEQ      +4
HLT      !DS
CMP      #-1, ARSWC
BEQ      +4
HLT      !WC
TST      ARSDA
BEQ      +4
HLT      !DA
CMP      @#0BUFSV, ARSBA
BEQ      +4
HLT      !BA
BIT      UNCMP, ARSAS
BNE      +4
HLT      !AS
CMP      #7, ARSER
BEQ      +4
HLT      !ER
CMP      #104220, ARSCS1
BEQ      +4
HLT      !DS

```



```

1612          ;*****
1613          ;TEST 45          TEST DRIVE CLEAR FUNCTION WITH ERRORS SET
1614          ;*****
1615 005752 104400 TST45: SCOPE
1616          ;FIRST SET ALL R/W BITS IN DISK REG
1617          ;DO DRIVE CLEAR-ALL R/W BITS SHOULD BE CLEARED
1618
1619 005754 104414 DRCLR: CLRDK          ;CLEAR ALL RS REG
1620 005756 012777 177777 173124 MOV      #177777,RSWC          ;LOAD RSWC
1621 005764 012777 177777 173122 MOV      #177777,RSDA          ;SET ALL POSSIBLE
1622 005772 012777 177017 173120 MOV      #177017,RSER          ;BITS IN DISK REG
1623 006000 012777 000070 173122 MOV      #70,RSMR          ;SET THESE BITS
1624 006006 012777 000011 173070 MOV      #11,RS1          ;SET DRIVE CLEAR
1625 006014 017700 173066 MOV      RS1,RS2,BAD          ;GET CS2 DATA
1626 006020 042700 177640 BIC      #177640,BAD          ;CLEAR JUNK
1627 006024 013701 001152 MOV      UNUM,GOOD          ;GET DRIVE UNIT
1628 006030 052701 000100 BIS      #100,GOOD          ;SET IR BIT
1629 006034 020100 CMP      GOOD,BAD          ;IS UNIT # THE SAME
1630 006036 001401 BEQ      .+4          ;YES
1631 006040 104000 HLT          ;UNIT # IN CS2 GOT MODIFIED
1632 006042 005777 173046 TST      RS1          ;DID DA CLEAR
1633 006046 001401 BEQ      .+4          ;YES
1634 006050 104004 HLT          ;DA SHOULD BE 0
1635 006052 005777 173042 TST      RSER          ;DID ER CLEAR
1636 006056 001401 BEQ      .+4          ;YES
1637 006060 104002 HLT          ;ER SHOULD BE CLEARED
1638 006062 017737 173042 001174 MOV      RS1,WORK          ;GET MR REG
1639 006070 042737 177707 001174 BIC      #177707,WORK          ;CLEAR JUNK
1640 006076 022737 000070 001174 CMP      #70,WORK          ;IS 70 STILL SET IN MR?
1641 006104 001401 BEQ      .+4          ;YES
1642 006106 104220 HLT          ;BITS 70 SHOULD NOT CLEAR
1643 006110 022777 004210 172766 CMP      #4210,RS1          ;DID THESE BITS CLEAR?
1644 006116 001401 BEQ      .+4          ;YES
1645 006120 104001 HLT          ;CS1 SHOULD =4210
1646 006122 033777 001156 172772 BIT      UNCMP,RSAS          ;AS SHOULD NOT SET
1647 006130 001401 BEQ      .+4          ;ON A DRIVE CLEAR FUN
1648 006132 104100 HLT          ;WHY DID AS SET?
1649 006134 022777 177777 172746 CMP      #177777,RSWC          ;DID RSWC CHANGE?
1650 006142 001401 BEQ      .+4          ;NO
1651 006144 104010 HLT          ;WC

```



```

1652 ;DO ONE WORD WRITE
1653 ;*****
1654 ;TEST 46 EXECUTE THE ONE WORD WRITE
1655 ;*****
1656 006146 104400 TST46: SCOPE
1657
1658 006150 104414 WRTST: CLRDK
1659 006152 012737 177777 027364 MOV #177777,OUTBUF ;CLEAR ALL RS REG
1660 006160 013777 001102 172724 MOV @#0BUFSV,@RSBA ;DATA TO BE X-FERED
1661 006166 012777 177777 172714 MOV #-1,@RSWC ;SET UP CURRENT ADDRESS
1662 006174 012777 000060 172702 1S: MOV #60,@RSCS1 ;SET WORD COUNT TO -1
1663 006202 022777 177777 172700 CMP #-1,@RSWC ;SET FUNCTION WITH NO GO BIT
1664 006210 001401 BEQ +4 ;DID WC MOVE?
1665 006212 104010 HLT !WC ;NO
1666 006214 023777 001102 172670 CMP @#0BUFSV,@RSBA ;WC MOVED
1667 006222 001401 BEQ +4 ;DID RSBA MOVE?
1668 006224 104020 HLT !BA ;NO
1669 006226 052777 000001 172650 BIS #BIT0,@RSCS1 ;BA MOVED
1670 006234 105777 172644 2S: TSTB @RSCS1 ;SET GO BIT
1671 006240 100001 BPL +4 ;TEST FOR RDY=0
1672 006242 104001 HLT !CS1 ;RDY=0
1673 006244 004737 026600 JSR PC, WAITRY ;RDY SHOULD = 0
1674 006250 104001 HLT !CS1 ;WAIT FOR READY
1675 006252 022777 000001 172634 CMP #1,@RSDA ;SHOULD = 260 RDY NEVER CAME UP
1676 006260 001401 BEQ +4 ;IS RSDA CORRECT
1677 006262 104004 HLT !DA ;RSDA OK
1678 006264 022777 004260 172612 3S: CMP #4260,@RSCS1 ;SHOULD = 1 SHOULD INCREMENT
1679 006272 001401 BEQ +4 ;IS ERROR FLAG SET?
1680 006274 104047 HLT !CS1!ER!DS!DA ;NO! X-FER OK
1681 006276 005777 172606 4S: TST @RSWC ;ERROR DURING X-FER
1682 006302 001401 BEQ +4 ;FETCH WORD COUNT
1683 006304 104010 HLT !WC ;WORD COUNT DID OVERFLOW
1684 006306 022777 010600 172602 CMP #10600,@RSDS ;SHOULD = 0 FAILED TO INCREMENT
1685 006314 001401 BEQ +4 ;IS RSDS OK?
1686 006316 104044 HLT !DS!DA ;YES
1687 006320 013701 001152 MOV UNNUM,GOOD ;NO
1688 006324 052701 000100 BIS #100,GOOD ;GET UNIT #
1689 006330 017700 172552 MOV @RSCS2,BAD ;SET IR BIT
1690 006334 020100 CMP GOOD,BAD ;GET CS2
1691 006336 001401 BEQ +4 ;IS CS2 CORRECT?
1692 006340 104000 HLT ;YES
1693 006342 017700 172544 MOV @RSBA,BAD ;BAD = CS2 GOOD IS CORRECT ANS
1694 006346 013701 001102 MOV @#0BUFSV,GOOD ;GET BA DATA
1695 006352 062701 000002 ADD #2,GOOD ;WHAT RSBA SHOULD EQUAL
1696 006356 020100 CMP GOOD,BAD ;UPDATE OUTBUFFER
1697 006360 001401 BEQ +4 ;IS RSBA CORRECT
1698 006362 104000 HLT ;YES
1699 006364 005777 172530 TST @RSER ;BA FAILED TO INCREMENT
1700 006370 001401 BEQ +4 ;DID ANY ERRORS SET?
1701 006372 104040 HLT !DS ;NO

```



```

1702 ;TEST READ FUNCTION
1703
1704 ;*****
1705 ;TEST 47 EXECUTE THE ONE WORD READ
1706 ;*****
1707 006374 104400 TST47: SCOPE
1708
1709 006376 104414 RDTST: CLRDK ;CLEAR ALL RS REG
1710 006400 005037 027364 CLR ;CLR TO READ INTO
1711 006404 013777 001102 172500 MOV ;SET UP CURRENT ADDRESS
1712 006412 012777 177777 172470 MOV ;SET WORD COUNT TO -1
1713 006420 012777 000071 172456 1S: MOV ;GO READ
1714 006426 105777 172452 2S: TSTB ;TEST FOR BUSY=1
1715 006432 100001 BPL ;BUSY SET
1716 006434 104001 HLT ;BUSY NOT SET
1717 006436 004737 026600 JSR ;WAIT FOR READY
1718 006442 104001 HLT ;TIMEOUT RDY DID NOT SET
1719 006444 022777 000001 172442 CMP ;WAS RSDA INCREMENTED BY 1
1720 006452 001401 BEQ ;RSDA OK
1721 006454 104046 HLT ;RSDA SHOULD CONTAIN A 1
1722 006456 022777 004270 172420 3S: CMP ;IS ERROR FLAG SET?
1723 006464 001401 BEQ ;NO! X-FER OK
1724 006466 104043 HLT ;RSCS1 SHOULD = 270
1725 006470 005777 172414 4S: TST ;TEST WC
1726 006474 001401 BEQ ;WORD COUNT DID OVERFLOW
1727 006476 104010 HLT ;SHOULD = 0
1728 006500 013701 001152 MOV ;GET CORRECT
1729 006504 052701 000100 BIS ;ANS OF CS2
1730 006510 017700 172372 MOV ;GET CS2
1731 006514 020100 CMP ;IS CS2 CORRECT?
1732 006516 001401 BEQ ;YES
1733 006520 104000 HLT ;GOOD = CORRECT ANS FOR CS2
1734 006522 017700 172364 MOV ;FETCH CURRENT ADDRESS
1735 006526 013701 001102 MOV ;WHAT RSBA SHOULD EQUAL
1736 006532 062701 000002 ADD ;UPDATE IT
1737 006536 020001 CMP ;IS RSBA CORRECT
1738 006540 001401 BEQ ;YES EXECUTE CONTINUE
1739 006542 104000 HLT ;RSBA FAILED TO INCREMENT
1740 006544 013700 027364 MOV ;GET DATA READ FROM DISK
1741 006550 012701 177777 MOV ;GET CORRECT ANS
1742 006554 020100 CMP ;IS OUTBUF CORRECT
1743 006556 001401 BEQ ;YES
1744 006560 104000 HLT ;GOOD=CORRECT ANS BAD=DATA READ FROM DISK

```


H04

MAINDEC-11-DZRSB-E
DZRSBE.P11 TST50RH11-RS03LA-RS03-RS04 BASIC FUNCTION DIAGNOSTIC MACY11 27(732) 25-SEP-76 09:06 PAGE 47
TEST WRITE CHECK

```

1745 ;*****
1746 ;TEST 50 TEST WRITE CHECK
1747 ;*****
1748 006562 104400 TST50: SCOPE
1749 ;DO A ONE WORD WRITE CHECK
1750
1751 ;* * *EXECUTE THE ONE WORD WRITE CHECK* * *
1752
1753 006564 104414 WRCKT: CLRDK ;CLEAR ALL RS REG
1754 006566 012737 177777 027364 MOV #177777,OUTBUF ;DATA TO BE X-FERED
1755 006574 013777 001102 172310 MOV @#0BUFSV,@RSBA ;SET UP CURRENT ADDRESS
1756 006602 012777 177777 172300 MOV #-1,@RSWC ;SET WORD COUNT TO -1
1757 006610 012777 000051 172266 1$: MOV #51,@RSCS1 ;GO WRITE CHECK
1758 006616 105777 172262 2$: TSTB @RSCS1 ;TEST FOR READY
1759 006622 100001 BPL +4 ;NOT READY
1760 006624 104001 HLT !CS1 ;BUSY FAILED TO SET
1761 006626 004737 026600 RSWCWT: JSR PC, WAITRY ;WAIT FOR READY
1762 006632 104001 HLT !CS1 ;BUSY FAILED TO CLEAR
1763 006634 013701 001152 MOV UNNUM,GOOD ;GET UNIT #
1764 006640 052701 000100 BIS #100,GOOD ;SET BIT IR
1765 006644 017700 172236 MOV @RSCS2,BAD ;GET CS2
1766 006650 020100 CMP GOOD,BAD ;IS CS2 CORRECT?
1767 006652 001401 BEQ +4 ;YES
1768 006654 104000 HLT ;GOOD = CORRECT ANS FOR CS2
1769 006656 022777 004250 172220 2$: CMP #4250,@RSCS1 ;ANY ERRORS?
1770 006664 001401 BEQ +4 ;X-FER OK
1771 006666 104046 HLT !DA!ER!DS ;ERROR DUR X-FER
1772 006670 022777 000001 172216 3$: CMP #BIT0,@RSDA ;WAS DAR INCREMENTED BY 1
1773 006676 001401 BEQ +4 ;RSDA OK
1774 006700 104004 HLT !DA ;DAR SHOULD = 1
1775 006702 005777 172202 TST @RSWC ;TEST FOR OVERFLOW
1776 006706 001401 BEQ +4 ;WORD COUNT DID OVERFLOW
1777 006710 104010 HLT !WC ;SHOULD = 0
1778 006712 017700 172174 MOV @RSBA,BAD ;FETCH CURRENT ADDRESS
1779 006716 013701 001102 MOV @#0BUFSV,GOOD ;WHAT RSBA SHOULD EQUAL
1780 006722 062701 000002 ADD #2,GOOD ;UPDATE IT
1781 006726 020001 CMP BAD,GOOD ;IS RSBA CORRECT
1782 006730 001401 BEQ +4 ;YES EXECUTE CONTINUE
1783 006732 104000 HLT ;RSBA FAILED TO INCREMENT

```



```

1784 ;DO ONE WORD WRITE ON -B- PORT
1785 ;IF A 1 WD TRANSFER KEEPS SETTING NEM PROGRAM WILL GO AND UPDATE
1786 ;ADDRESS (OBUFSV) ON -B- PORT BY 4K AND TRY TRANSFER AGAIN UNTILL IT
1787 ;REACHES 28K. IF NO TRANSFER IT THEN SKIPS WRITE,
1788 ;READ AND WRITE CHECK ON -B- PORT
1789 ;TO INHIBIT OBUFSV FROM CHANGING SET BIT 12
1790
1791 ;*****
1792 ;TEST 51 EXECUTE THE ONE WORD WRITE ON -B- PORT
1793 ;*****
1794 G06734 104400 TST51: SCOPE
1795
1796 006736 104414 WRTSTB: CLRDK
1797 006740 013737 001102 001174 MOV 2#OBUFSV,WORK ;CLEAR ALL RS REG
1798 006746 012777 177777 172220 MOV 2#177777,2#WORK ;GET LOC OF RSBA TO LOAD
1799 006754 013777 001102 172130 MOV 2#OBUFSV,2#RSBA ;DATA TO BE X-FERED
1800 006762 012777 177777 172120 MOV 2#-1,2#RSMC ;SET UP CURRENT ADDRESS
1801 006770 012777 002061 172106 MOV 2#2061,2#RSCS1 ;SET WORD COUNT TO -1
1802 006776 105777 172102 2$: TSTB 2#RSCS1 ;TEST B PORT
1803 007002 100001 BPL 2#4 ;TEST FOR RDY=0
1804 007004 104001 HLT 2#CS1 ;RDY=0
1805 007006 004737 026600 JSR PC,WAITRY ;RDY SHOULD = 0
1806 007012 104001 HLT 2#CS1 ;WAIT FOR READY
1807 007014 032777 010000 172004 BIT 2#BIT12,2#SWR ;SHOULD = 260 RDY NEVER CAME UP
1808 007022 001006 BNE 3$ ;INHIBIT ADDRESS?
1809 007024 032777 004000 172054 BIT 2#BIT11,2#RSCS2 ;YES
1810 007032 001402 BEQ 3$ ;DID NEM SET?
1811 007034 000137 025632 JMP FINDM ;NO
1812 007040 022777 000001 172046 3$: CMP 2#1,2#RSDA ;GO FIND MEMORY ON PORT B
1813 007046 001401 BEQ 2#4 ;IS RSDA CORRECT
1814 007050 104004 HLT 2#DA ;RSDA OK
1815 007052 022777 006260 172024 CMP 2#6260,2#RSCS1 ;SHOULD = 1 SHOULD INCREMENT
1816 007060 001401 BEQ 4$ ;IS CS1 CORRECT?
1817 007062 104001 HLT 2#CS1 ;YES
1818 007064 005777 172020 4$: TST 2#RSMC ;FETCH WORD COUNT
1819 007070 001401 BEQ 2#4 ;WORD COUNT DID OVERFLOW
1820 007072 104010 HLT 2#WC ;SHOULD = 0 FAILED TO INCREMENT
1821 007074 022777 010600 172014 CMP 2#10600,2#RSDS ;IS RSDS OK?
1822 007102 001401 BEQ 2#4 ;YES
1823 007104 104044 HLT 2#DS!DA ;NO
1824 007106 013701 001152 MOV UNNUM,GOOD ;GET UNIT #
1825 007112 052701 000100 BIS 2#100,GOOD ;SET IR BIT
1826 007116 017700 171764 MOV 2#RSCS2,BAD ;GET CS2
1827 007122 020100 CMP GOOD,BAD ;IS CS2 CORRECT?
1828 007124 001401 BEQ 2#4 ;YES
1829 007126 104000 HLT ;BAD = CS2 GOOD IS CORRECT ANS
1830 007130 017700 171756 MOV 2#RSBA,BAD ;GET BA DATA
1831 007134 013701 001102 MOV 2#OBUFSV,GOOD ;WHAT RSBA SHOULD EQUAL
1832 007140 062701 000002 ADD 2#2,GOOD ;UPDATE OUTBUFFER
1833 007144 020100 CMP GOOD,BAD ;IS RSBA CORRECT
1834 007146 001401 BEQ 2#4 ;YES
1835 007150 104000 HLT ;BA FAILED TO INCREMENT

```


J04

MAINDEC-11-DZRSB-E
DZRSBE.P11 TST52RH11-RS03LA-RS03-RS04 BASIC FUNCTION DIAGNOSTIC MACY11 27(732) 25-SEP-76 09:06 PAGE 49
EXECUTE THE ONE WORD READ ON -B- PORT

```

1836                                     ;*****
1837                                     ;TEST 52 EXECUTE THE ONE WORD READ ON -B- PORT
1838                                     ;*****
1839 007152 104400 TST52: SCOPE
1840
1841 007154 104414 RDTSTB: CLRDK
1842 007156 005037 001174 CLR WORK
1843 007162 013777 001102 171722 MOV @#OBUFSV, @RSBA
1844 007170 012777 177777 171712 MOV #-1, @RSWC
1845 007176 012777 002071 171700 MOV @2071, @RSCS1
1846 007204 105777 171674 25: TSTB @RSCS1
1847 007210 100001 BPL .+4
1848 007212 104001 HLT !CS1
1849 007214 004737 026600 JSR PC, WAITRY
1850 007220 104001 HLT !CS1
1851 007222 022777 000001 171664 CMP @BITO, @RSDA
1852 007230 001401 BEQ .+4
1853 007232 104046 HLT !DA!ER!DS
1854 007234 022777 006270 171642 CMP @6270, @RSCS1
1855 007242 001401 BEQ 45
1856 007244 104001 HLT !CS1
1857 007246 005777 171636 45: TST @RSWC
1858 007252 001401 BEQ .+4
1859 007254 104010 HLT !WC
1860 007256 013701 001152 MOV UNNUM, GOOD
1861 007262 052701 000100 BIS @100, GOOD
1862 007266 017700 171614 MOV @RSCS2, BAD
1863 007272 020100 CMP GOOD, BAD
1864 007274 001401 BEQ .+4
1865 007276 104000 HLT
1866 007300 017700 171606 MOV @RSBA, BAD
1867 007304 013701 001102 MOV @#OBUFSV, GOOD
1868 007310 062701 000002 ADD @2, GOOD
1869 007314 020001 CMP BAD, GOOD
1870 007316 001401 BEQ .+4
1871 007320 104000 HLT
1872 007322 013737 001102 001174 MOV @#OBUFSV, WORK
1873 007330 017700 171640 MOV @WORK, BAD
1874 007334 012701 177777 MOV #-1, GOOD
1875 007340 020100 CMP GOOD, BAD
1876 007342 001401 BEQ .+4
1877 007344 104000 HLT

```

```

;CLEAR ALL RS REG
;CLR TO READ INTO
;SET UP CURRENT ADDRESS
;SET WORD COUNT TO -1
;B PORT
;TEST FOR BUSY=1
;BUSY SET
;BUSY NOT SET
;WAIT FOR READY
;TIMEOUT RDY DID NOT SET
;WAS RSDA INCREMENTED BY 1
;RSDA OK
;RSDA SHOULD CONTAIN A 1
;TST B PORT
;OK
;CS1 SHOULD = 6270
;TEST WC
;WORD COUNT DID OVERFLOW
;SHOULD = 0
;GET CORRECT
;ANS OF CS2
;GET CS2
;IS CS2 CORRECT?
;YES
;GOOD = CORRECT ANS FOR CS2
;FETCH CURRENT ADDRESS
;WHAT RSBA SHOULD EQUAL
;UPDATE IT
;IS RSBA CORRECT
;YES EXECUTE CONTINUE
;RSBA FAILED TO INCREMENT
;GET DATA READ FROM DISK

;GET CORRECT ANS
;IS OUTBUF CORRECT
;YES
;GOOD=CORRECT ANS BAD=DATA READ FROM DISK

```


MAINDEC-11-DZRSB-E
DZRSBE.P11 TST53

RH11-RS03LA-RS03-RS04 BASIC FUNCTION DIAGNOSTIC MACY11 27(732) 25-SEP-76 09:06 PAGE 50
TEST WRITE CHECK ON -B- PORT

Line	Address	Hex	Hex	Hex	Hex	Instruction	Comment
1878							*****
1879						TEST 53	TEST WRITE CHECK ON -B- PORT
1880							*****
1881	007346	104400				TST53: SCOPE	
1882							
1883							
1884	007350	104414				WRCKTB: CLRDK	:CLEAR ALL RS REG
1885	007352	013737	001102	001174		MOV	:GET LOC FOR
1886	007360	012777	177777	171606		MOV	:DATA TO BE X-FERED
1887	007366	013777	001102	171516		MOV	:SET UP CURRENT ADDRESS
1888	007374	012777	177777	171506		MOV	:SET WORD COUNT TO -1
1889	007402	012777	02051	171474		MOV	:B PORT
1890	007410	105777	71470			25: TSTB	:TEST FOR READY
1891	007414	100001				BPL	:NOT READY
1892	007416	104001				HLT	:BUSY FAILED TO SET
1893	007420	004737	0.6600			JSR	:WAIT FOR READY
1894	007424	104001				HLT	:BUSY FAILED TO CLEAR
1895	007426	013701	001152			MOV	:GET UNIT #
1896	007432	052701	000100			BIS	:SET BIT IR
1897	007436	017700	171444			MOV	:GET CS2
1898	007442	020100				CMP	:IS CS2 CORRECT?
1899	007444	001401				BEQ	:YES
1900	007446	104000				HLT	:GOOD = CORRECT ANS FOR CS2
1901	007450	022777	006250	171426		CMP	:IS CS1 CORRECT?
1902	007456	001401				BEQ	:YES
1903	007460	104001				HLT	:CS1 SHOULD = 6250
1904	007462	022777	000001	171424		35: CMP	:WAS DAR INCREMENTED BY 1
1905	007470	001401				BEQ	:RSDA OK
1906	007472	104004				HLT	:DAR SHOULD = 1
1907	007474	005777	171410			TST	:TEST FOR OVERFLOW
1908	007500	001401				BEQ	:WORD COUNT DID OVERFLOW
1909	007502	104010				HLT	:SHOULD = 0
1910	007504	017700	171402			MOV	:FETCH CURRENT ADDRESS
1911	007510	013701	001102			MOV	:WHAT RSBA SHOULD EQUAL
1912	007514	062701	000002			ADD	:UPDATE IT
1913	007520	020001				CMP	:IS RSBA CORRECT
1914	007522	001401				BEQ	:YES EXECUTE CONTINUE
1915	007524	104000				HLT	:RSBA FAILED TO INCREMENT

L04

MAINDEC-11-DZRSB-E
DZRSBE.P11

TST53

RH11-RS03LA-RS03-RS04 BASIC FUNCTION DIAGNOSTIC
TEST WRITE CHECK ON -B- PORT

MACY11 27(732) 25-SEP-76 09:06 PAGE 51

```

1916 007526 013737 001102 001170 NXM: MOV OBUFSV,BPORTT ;SAVE -B- PORT BUFFER
1917 007534 012737 027364 001102 MOV #OUTBUF,OBUFSV ;RESTORE OBUFSV
1918
1919 ;DESELECT THEN SELECT UNIT NUMBER IN RSCS2 CHECK TIMING
1920 *****
1921 ;TEST 54 DESELECT THEN SELECT UNIT NUMBER TIMING TEST
1922 *****
1923 007542 104400 TST54: SCOPE
1924
1925 007544 104414 UNITST: CLRDK
1926 007546 005004 CLR R4 ;CLEAR ALL RS REG
1927 007550 020437 001152 CMP R4,UNNUM ;CLEAR R4
1928 007554 001001 BNE 3$ ;IS THIS CORRECT UNIT #?
1929 007556 005204 INC R4 ;NO THEN USE IT
1930 007560 012737 177777 027364 3$: MOV #177777,OUTBUF ;GET WRONG DRIVE
1931 007566 013777 001102 171316 MOV #OBUFSV,RSBA ;DATA TO BE X-FERED
1932 007574 012777 177777 171306 MOV #-1,RSWC ;SET UP CURRENT ADDRESS
1933 007602 012703 000061 MOV #61,R3 ;SET WORD COUNT TO -1
1934 007606 013705 001152 MOV UNNUM,R5 ;GET WRITE FUNCTION
1935 007612 012701 172040 MOV #172040,R1 ;GET CORRECT UNIT #
1936 007616 010461 000010 MOV R4,10(R1) ;GET CS1 REG
1937 007622 000240 NOP ;LOAD WRONG UNIT # INTO CS2
1938 007624 010561 000010 MOV R5,10(R1) ;WAIT FOR DRIVE TO SETTLE
1939 007630 010311 1$: MOV R3,(R1) ;LOAD CORRECT UNIT #
1940 007632 004737 026600 JSR PC,WAITRY ;LOAD FUNCTION IN CS1
1941 007636 104001 HLT !CS1 ;WAIT FOR READY
1942 007640 022777 004260 171236 CMP #4260,RSRCS1 ;SHOULD = 260 RDY NEVER CAME UP
1943 007646 001401 BEQ .+4 ;IS ERROR FLAG SET?
1944 007650 104047 HLT !CS1!ER!DS!DA ;NO! X-FER OK
1945 007652 022777 000001 171234 CMP #1,RSRSDA ;ERROR DURING X-FER
1946 007660 001401 BEQ .+4 ;IS RSDA CORRECT
1947 007662 104004 HLT !DA ;RSDA OK
1948 007664 005777 171220 TST RSWC ;SHOULD = 1 SHOULD INCREMENT
1949 007670 001401 BEQ .+4 ;FETCH WORD COUNT
1950 007672 104010 HLT !WC ;WORD COUNT DID OVERFLOW
1951 007674 022777 010600 171214 CMP #10600,RSRSDS ;SHOULD = 0 FAILED TO INCREMENT
1952 007702 001401 BEQ .+4 ;IS RSDS OK?
1953 007704 104044 HLT !DS!DA ;YES
1954 007706 013701 001152 MOV UNNUM,GOOD ;NO
1955 007712 052701 000100 BIS #100,GOOD ;GET UNIT #
1956 007716 017700 171164 MOV RSCS2,BAD ;SET IR BIT
1957 007722 020100 CMP GOOD,BAD ;GET CS2
1958 007724 001401 BEQ .+4 ;IS CS2 CORRECT?
1959 007726 104000 HLT ;YES
1960 007730 017700 171156 MOV RRSBA,BAD ;BAD = CS2 GOOD IS CORRECT ANS
1961 007734 013701 001102 MOV #OBUFSV,GOOD ;GET BA DATA
1962 007740 062701 000002 ADD #2,GOOD ;WHAT RSBA SHOULD EQUAL
1963 007744 020100 CMP GOOD,BAD ;UPDATE OUTBUFFER
1964 007746 001401 BEQ .+4 ;IS RSBA CORRECT
1965 007750 104000 HLT ;YES
;BA FAILED TO INCREMENT

```


M04

MAINDEC-11-DZRSB-E
DZRSBE.P11 TST54RH11-RS03LA-RS03-RS04 BASIC FUNCTION DIAGNOSTIC MACY11 27(732) 25-SEP-76 09:06 PAGE 52
DESELECT THEN SELECT UNIT NUMBER TIMING TEST

```

1966 ;TEST CURRENT ADDRESS INHIBT-BAI IN RSCS2
1967 ;DO A ONE WORD WRITE AND SEE
1968 ;IF RSBA INCREMENTED AFTER THE X-FER
1969
1970 *****
1971 ;TEST 55 TEST BAI IN RSCS2
1972 *****
1973 TST55: SCOPE
1974
1975 007752 104400 BAITST: CLRDK
1976 007754 104414
1977 007756 013777 001102 171126 MOV @#OBUFSV, @RSBA ;CLEAR ALL RS REG
1978 007764 012777 177777 171116 MOV #-1, @RSWC ;SET UP CURRENT ADDR
1979 007772 052777 000010 171106 BIS #BAI, @RSCS2 ;SET WORD COUNT TO -1
1980 010000 012777 000061 171076 MOV #61, @RSCS1 ;SET BAI BIT
1981 010006 004737 026600 JSR PC, WAITRY ;WRITE
1982 010012 104001 HLT !CS1 ;WAIT FOR READY
1983 010014 013701 001102 171066 1$: MOV @#OBUFSV, GOOD ;RDY DID NOT SET
1984 010020 017700 MOV @RSBA, BAD ;WHAT RSBA SHOULD BE
1985 010024 020100 CMP GOOD, BAD ;WHAT RSBA IS
1986 010026 001401 BEQ .+4 ;COMPARE
1987 010030 104000 HLT ;YES
1988 010032 005777 171062 TST @RSER ;BAD=OUTBUF GOOD = CORRECT ANS
1989 010036 001401 BEQ .+4 ;ANY ERRORS?
1990 010040 104040 HLT !DS ;NO
1991 010042 104414 CLRDK ;YES
1992 010044 032777 000010 171034 BIT #BAI, @RSCS2 ;CLEAR ALL RS REG
1993 010052 001401 BEQ .+4 ;DID BAI CLEAR?
1994 010054 104002 HLT !ER ;YES
1995 ;BAI DID NOT SET

```


N04

MAINDEC-11-DZRSB-E
DZRSBE.P11 TST56

RH11-R503LA-R503-R504 BASIC FUNCTION DIAGNOSTIC MACY11 27(732) 25-SEP-76 09:06 PAGE 53
TEST NON-EXISTENT MEMORY ERROR BIT IN CS2

```

1994
1995
1996
1997 010056 104400
1998
1999 010060 104414
2000 010062 052777 000010 171016
2001 010070 012777 177600 171012
2002 010076 012777 173000 171006
2003 010104 012777 001471 170772
2004 010112 004737 026600
2005 010116 104040
2006 010120 013701 001152
2007 010124 052701 004310
2008 010130 017700 170752
2009 010134 020100
2010 010136 001401
2011 010140 104000
2012 010142 022777 145670 170734
2013 010150 001401
2014 010152 104001
2015 010154 012777 040000 170722
2016 010162 017700 170720
2017 010166 013701 001152
2018 010172 052701 000310
2019 010176 020100
2020 010200 001401
2021 010202 104200
2022
2023
2024
2025
2026 010204 104400
2027
2028 010206 104414
2029 010210 012777 000032 170676
2030 010216 013777 001102 170666
2031 010224 012777 177777 170656
2032 010232 005037 001174
2033 010236 032777 001000 170664 45:
2034 010244 001004
2035 010246 005337 001174
2036 010252 001371
2037 010254 104220

```

TEST 56 TEST NON-EXISTENT MEMORY ERROR BIT IN CS2

TST56: SCOPE

NXMTSM: CLRDK
BIS #BAI, ARSCS2
MOV #-200, ARSWC
MOV #173000, ARSBA
MOV #1471, ARSCS1
JSR PC, WAITRY
HLT !DS
TSTNEM: MOV UNNUM, GOOD
BIS #4310, GOOD
MOV ARSCS2, BAD
CMP GOOD, BAD
BEQ .+4
HLT
CMP #145670, ARSCS1
BEQ .+4
HLT !CS1
MOV #TRE, ARSCS1
MOV ARSCS2, BAD
MOV UNNUM, GOOD
BIS #310, GOOD
CMP GOOD, BAD
BEQ .+4
HLT !CS2

; CLEAR ALL RS REG
; SET BAI BIT
; SET UP WORD COUNT
; SET UP CURRENT ADDRESS
; READ AND LOAD A16 +A17 FOR 18 BIT ADDRESS
; WAIT FOR READY
; READY NEVER CAME UP
; GET UNIT NO.
; SET BAI+OR BITS
; GET CS2
; IS CS2 CORRECT?
; YES
; BAD=CS2 GOOD=CORRECT ANS FOR CS2
; DID TRE SET?
; YES
; TRE SHOULD SET BECAUSE OF NEM
; CLEAR TRE
; GET CS2
; GET DRIVE
; SET IR
; IS CS2 CORRECT?
; YES
; CS2=BAD GOOD IS CORRECT ANS FOR CS2

TEST 57 TEST BLOCK SEARCH FUNCTION, PIP AND DRY BIT AND ADDR. CONF BIT

TST57: SCOPE

BLOCK: CLRDK
MOV #32, ARSDA
MOV #08UFSV, ARSBA
MOV #-1, ARSWC
CLR WORK
BIT #1000, ARSMR
BNE 35
DEC WORK
BNE 45
HLT !MR

; CLEAR ALL RS REG
; DO A SEARCH FOR SECTOR 32
; LOAD REGS. TO MAKE
; SURE THEY DO NOT CHANGE
; SETUP FOR TIMEOUT ROUTINE
; WAIT FOR DISK TO
; REACH SECTOR 32
; TIME OUT
; ROUTINE
; COULD NOT FIND SECTOR 32

MAINDEC-11-DZRSB-E
DZRSBE.P11

TST57

RH11-RS03LA-RS03-RS04 BASIC FUNCTION DIAGNOSTIC
TEST BLOCK SEARCH FUNCTION, PIP AND DRY BIT ANDMACY11 27(732) 25-SEP-76 09:06 PAGE 54
ADDR. CONF BIT

2038	010256	005077	170632		3\$:	CLR	DRSDA	:NOW SEARCH FOR 0
2039	010262	012777	000031	170614		MOV	#31,DRSCS1	:DO A BLOCK SEARCH FUNCTION
2040	010270	032777	000200	170620		BIT	#DRY,DRSDS	:IS DRY CLEARED?
2041	010276	001402				BEQ	1\$	:YES
2042	010300	104040				HLT	!DS	:DRY SHOULD BE CLEARED DURING A BLOCK SEARCH
2043	010302	000500				BR	00UT	:GET OUT BECAUSE OF TIMING
2044	010304	032777	020000	170604	1\$:	BIT	#20000,DRSDS	:IS PIP SET?
2045	010312	001001				BNE	+4	:YES
2046	010314	104040				HLT	!DS	:PIP SHOULD BE SET
2047	010316	012701	020000			MOV	#20000,GOOD	:SETUP FOR TIMEOUT ROUTINE
2048	010322	005301			2\$:	DEC	GOOD	:DO TIMEOUT
2049	010324	001466				BEQ	TTMOUT	:TIMED OUT
2050	010326	032777	000200	170562		BIT	#DRY,DRSDS	:DID DRY SET?
2051	010334	001772				BEQ	2\$	:NO
2052	010336	022777	110600	170552		CMP	#110600,DRSDS	:DID PIP CLEAR?
2053	010344	001401				BEQ	+4	:YES
2054	010346	104040				HLT	!DS	:PIP BIT DID NOT CLEAR
2055	010350	022777	104230	170526		CMP	#104230,DRSCS1	:DID SC SET?
2056	010356	001401				BEQ	+4	:YES
2057	010360	104041				HLT	!CS1!DS	:SC DID NOT SET
2058	010362	013737	001152	001174		MOV	UNNUM,WORK	:GET CORRECT AS BIT
2059	010370	005001				CLR	GOOD	:IN RSAS REG
2060	010372	000261				SEC		:THAT SHOULD
2061	010374	006101			5\$:	ROL	GOOD	:BE SET
2062	010376	005737	001174			TST	WORK	
2063	010402	001403				BEQ	6\$	
2064	010404	005337	001174			DEC	WORK	
2065	010410	000771				BR	5\$	
2066	010412	020177	170504		6\$:	CMP	GOOD,DRSAS	:IS RSAS CORRECT?
2067	010416	001403				BEQ	7\$	:YES
2068	010420	017700	170476			MOV	DRSAS,BAD	:NO
2069	010424	104000				HLT		
2070	010426	010177	170470		7\$:	MOV	GOOD,DRSAS	:CLEAR AS REG
2071	010432	005777	170464			TST	DRSAS	:DID IT CLEAR?
2072	010436	001401				BEQ	+4	:YES
2073	010440	104100				HLT	!AS	:NO
2074	010442	022777	010600	170446		CMP	#10600,DRSDS	:DID ATA CLEAR?
2075	010450	001401				BEQ	+4	:YES
2076	010452	104040				HLT	!DS	:NO
2077	010454	023777	001102	170430		CMP	#30BUFSV,DRSBA	:DID BA MOVE?
2078	010462	001401				BEQ	+4	:NO
2079	010464	104021				HLT	!CS1!BA	:BA MOVED WHY?
2080	010466	022777	177777	170414		CMP	#-1,DRSMC	:DID WC MOVE?
2081	010474	001401				BEQ	+4	:NO
2082	010476	104010				HLT	!WC	:WC MOVED WHY?
2083	010500	000401				BR	00UT	:DONE GET OUT
2084	010502	104040				HLT	!DS	:DYR NEVER CAME UP
2085	010504							:DONE

TTMOUT:
00UT:


```

2086 ;*****
2087 ;TEST 60 ILLEGAL FUNCTION CODE TEST CODE 3 TO 51
2088 ;*****
2089 010504 104400 TST60: SCOPE
2090
2091 ;TEST ILF BIT IN RSER AND ERR BIT IN RSDS
2092 ;ALSO CHECKS TO SEE IF WC,BA, OR DA GOT MODIFIED
2093 ;IF WISHING TO LOOP ON ONE FUNCTION ONLY, LOAD
2094 ;FUNCTION INTO LOCATION ILLTAB: AND 0 IN FOLLOWING LOCATION
2095
2096 010506 013737 022414 001164 ILL51: MOV TIMES,TIMSV ;SAVE LOOP COUNT
2097 010514 012737 000010 022414 MOV #10,TIMES ;LOOP TEN TIMES
2098 010522 104414 CLRDK ;CLEAR ALL RS REG
2099 010524 012703 027300 1$: MOV #ILLTAB,R3 ;GET STARTING ADD OF TABLE
2100 010530 012300 3$: MOV (R3)+,BAD ;GET ILL FUN
2101 010532 001513 BEQ ILFDN ;DONE GET OUT
2102 010534 013777 001102 170350 MOV #0BUFSV,RSBA ;SET UP REGS.
2103 010542 012777 177777 170340 MOV #-1,RSWC ;TO CHECK FOR CHANGE
2104 010550 010077 170330 2$: MOV BAD,RS1 ;DO ILLEGAL FUNCTION
2105 010554 042700 000001 BIC #BIT0,BAD ;CLEAR GO BIT
2106 010560 010001 MOV BAD,GOOD ;MOV ILLEGAL FUN INTO GOOD
2107 010562 105777 170316 6$: TSTB RS1 ;RDY SET?
2108 010566 100375 BPL 6$ ;NO
2109 010570 052701 104200 BIS #104200,GOOD ;SET ERROR BITS
2110 010574 017700 170304 4$: MOV RS1,BAD ;PUT CS1 INTO BAD
2111 010600 020100 CMP GOOD,BAD ;IS CS1 CORRECT?
2112 010602 001401 BEQ .+4 ;YES
2113 010604 104000 HLT ;GOOD IS WHAT CS1 SHOULD =BAD=CS1
2114 010606 022777 000001 170304 CMP #1,RSER ;DID ILF SET?
2115 010614 001401 BEQ .+4 ;YES
2116 010616 104043 HLT ;ILF DID NOT SET
2117 010620 022777 150600 170270 CMP #150600,RS1 ;IS DS GOOD?
2118 010626 001401 BEQ .+4 ;YES
2119 010630 104043 HLT ;ERR DID NOT SET
2120 010632 017700 170250 MOV RS2,BAD ;GET CS2
2121 010636 013701 001152 MOV UNUM,GOOD ;GET UNIT #
2122 010642 052701 000100 BIS #100,GOOD ;SET IR BIT
2123 010646 020100 CMP GOOD,BAD ;IS CS2 CORRECT?
2124 010650 001401 BEQ .+4 ;YES
2125 010652 104000 HLT ;GOOD = CORRECT ANS FOR CS2
2126 010654 013701 001156 MOV UNCM,GOOD ;GET CORRECT DRIVE
2127 010660 042701 177400 BIC #177400,GOOD ;CLEAR UNWANTED BITS
2128 010664 017700 170232 MOV RSAS,BAD ;GET RSAS REG
2129 010670 020001 CMP BAD,GOOD ;DID CORRECT UNIT ANSWER?
2130 010672 001401 BEQ .+4 ;YES
2131 010674 104100 HLT ;NO WRONG DRIVE ANSWERED

```


MAINDEC-11-DZRSB-E
DZRSBE.P11

TST60

RH11-RS03LA-RS03-RS04
ILLEGAL FUNCTION CODEBASIC FUNCTION DIAGNOSTIC MACY11 27(732) 25-SEP-76 09:06 PAGE 56
TEST CODE 3 TO 51

2132	010676	023777	001102	170206	CMP	#0BUFSV, ARSBA	:DID BA MOVE
2133	010704	001401			BEQ	+4	:NO
2134	010706	104021			HLT	:CS1!BA	:BA MOVED ON AN ILLEGAL FUNCTION
2135	010710	022777	177777	170172	CMP	#-1, ARSWC	:DID WC MOVE?
2136	010716	001401			BEQ	+4	:NO
2137	010720	104011			HLT	:CS1!WC	:WC MOVED
2138	010722	005777	170166		TST	ARSDA	:DID DA MOVE
2139	010726	001401			BEQ	+4	:NO
2140	010730	104005			HLT	:CS1!DA	:DA MOVED
2141	010732	104414			CLRDK		:CLEAR ALL ERRORS
2142	010734	005777	170160		TST	ARSR	:DID ERRORS CLEAR
2143	010740	001401			BEQ	+4	:YES
2144	010742	104040			HLT	:DS	:ILF DID NOT CLEAR
2145	010744	022777	004200	170132	CMP	#4200, ARSCS1	:DID ERRORS IN CS1 CLEAR
2146	010752	001401			BEQ	+4	:YES
2147	010754	104040			HLT	:DS	
2148	010756	000137	010530		JMP	3\$	:CONTINUE UNTIL DONE
2149	010762						:DONE WITH ILLEGAL FUNCTION TEST
2150					ILFDN:		
2151					:*****		
2152					:TEST 61	ILLEGAL FUNCTION CODE TEST CODE 53 TO 77	
2153	010762	104400			:*****		
2154					TST61: SCOPE		
2155					:TEST ILF BIT IN RSER AND ERR BIT IN RSDS		
2156					:ALSO CHECKS TO SEE IF WC, BA, OR DA GOT MODIFIED		
2157					:IF WISHING TO LOOP ON ONE FUNCTION ONLY, LOAD		
2158					:FUNCTION INTO LOCATION ILFTB2: AND 0 IN FOLLOWING LOCATION		
2159							
2160	010764	104414			ILLFUN: CLRDK		:CLEAR ALL RS REG
2161	010766	012703	027342		1\$: MOV	#ILFTB2, R3	:GET TABLE OF ILL FUNS.
2162	010772	005037	001174		3\$: CLR	WORK	:CLEAR WORK
2163	010776	012300			MOV	(R3)+, BAD	:GET ILL FUN
2164	011000	001554			BEQ	ILFDNE	:DONE GET OUT
2165	011002	013777	001102	170102	MOV	#0BUFSV, ARSBA	:SET UP REGS.
2166	011010	012777	177777	170072	MOV	#-1, ARSWC	:TO CHECK FOR CHANGE
2167	011016	010037	001176		MOV	BAD, WORK1	:SHOULD WE TEST
2168	011022	042737	177707	001176	BIC	#177707, WORK1	:BA AND WC
2169	011030	022737	000060	001176	CMP	#60, WORK1	:TO INC
2170	011036	001003			BNE	2\$	:NO
2171	011040	012737	000007	001174	MOV	#7, WORK	:YES
2172	011046	010077	170032		2\$: MOV	BAD, ARSCS1	:DO ILLEGAL FUNCTION
2173	011052	042700	000001		10\$: BIC	#BIT0, BAD	:CLEAR GO BIT
2174	011056	010001			MOV	BAD, GOOD	:MOV ILLEGAL FUN INTO GOOD

E05

MAINDEC-11-DZRSB-E
DZRSBE.P11

TST61

RH11-RS03LA-RS03-RS04 BASIC FUNCTION DIAGNOSTIC MACY11 27(732) 25-SEP-76 09:06 PAGE 57
ILLEGAL FUNCTION CODE TEST CODE 53 TO 77

2175	011060	105777	170020	6S:	TSTB	ARSCS1	:RDY SET?
2176	011064	100375			BPL	6S	:NO
2177	011066	052701	144200		BIS	#144200,GOOD	:SET ERROR BITS
2178	011072	017700	170006		MOV	ARSCS1,BAD	:PUT CS1 INTO BAD
2179	011076	020100			CMP	GOOD,BAD	:IS CS1 CORRECT?
2180	011100	001401			BEQ	.+4	:YES
2181	011102	104000			HLT		:GOOD IS WHAT CS1 SHOULD =BAD=CS1
2182	011104	022777	000001 170006		CMP	#1,ARSER	:DID ILF SET?
2183	011112	001401			BEQ	.+4	:YES
2184	011114	104043			HLT	:CS1!ER!DS	:ILF DID NOT SET
2185	011116	022777	150600 167772		CMP	#150600,ARSDS	:IS DS GOOD?
2186	011124	001401			BEQ	.+4	:YES
2187	011126	104043			HLT	:CS1!ER!DS	:ERR DID NOT SET
2188	011130	005777	167760		TST	ARSDA	:DID DA MOVE?
2189	011134	001401			BEQ	.+4	:NO
2190	011136	104005			HLT	:CS1!DA	:DA MOVED
2191	011140	005737	001174		TST	WORK	:IS THIS AN ILL WRITE FUN?
2192	011144	001025			BNE	11S	:YES
2193	011146	017700	167734		MOV	ARSCS2,BAD	:GET CS2
2194	011152	013701	001152		MOV	UNNUM,GOOD	:GET UNIT #
2195	011156	052701	001100		BIS	#1100,GOOD	:SET IR BIT
2196	011162	020100			CMP	GOOD,BAD	:IS CS2 CORRECT?
2197	011164	001401			BEQ	.+4	:YES
2198	011166	104000			HLT		:GOOD = CORRECT ANS FOR CS2
2199	011170	023777	001102 167714		CMP	#0BUFSV,ARSDA	:DID BA MOVE
2200	011176	001401			BEQ	.+4	:NO
2201	011200	104021			HLT	:CS1!BA	:BA MOVED ON AN ILLEGAL FUNCTION
2202	011202	022777	177777 167700		CMP	#-1,ARSWC	:DID WC MOVE?
2203	011210	001401			BEQ	.+4	:NO
2204	011212	104011			HLT	:CS1!WC	:WC MOVED
2205	011214	000137	011302		JMP	4S	:CONTINUE UNTIL DONE
2206	011220	017700	167662	11S:	MOV	ARSCS2,BAD	:GET CS2
2207	011224	013701	001152		MOV	UNNUM,GOOD	:GET UNIT #
2208	011230	052701	001300		BIS	#1300,GOOD	:SET IR BIT
2209	011234	020100			CMP	GOOD,BAD	:IS CS2 CORRECT?
2210	011236	001401			BEQ	.+4	:YES
2211	011240	104000			HLT		:GOOD = CORRECT ANS FOR CS2
2212	011242	013737	001102 001174		MOV	#0BUFSV,WORK	:GET BUFFER ADDR.
2213	011250	062737	000002 001174		ADD	#2,WORK	:UPDATE IT
2214	011256	023777	001174 167626		CMP	WORK,ARSDA	:DID BA MOVE
2215	011264	001401			BEQ	.+4	:YES
2216	011266	104021			HLT	:CS1!BA	:BA MOVED ON AN ILLEGAL FUNCTION
2217	011270	022777	000000 167612		CMP	#0,ARSWC	:DID WC MOVE?
2218	011276	001401			BEQ	.+4	:NO
2219	011300	104011			HLT	:CS1!WC	:WC MOVED
2220	011302	104414		4S:	CLDK		:CLEAR ALL ERRORS
2221	011304	022777	004200 167572		CMP	#4200,ARSCS1	:DID ERRORS CLEAR
2222	011312	001401			BEQ	.+4	:YES
2223	011314	104040			HLT	:DS	:NO
2224	011316	005777	167576		TST	ARSER	:DID ERROR CLEAR
2225	011322	001401			BEQ	.+4	:NO
2226	011324	104040			HLT	:DS	:YES
2227	011326	000137	010772		JMP	3S	:CONTINUE UNTIL DONE
2228	011332			ILFDNE:			:DONE WITH ILLEGAL FUNCTION TEST

F05

MAINDEC-11-DZRSB-E
DZRSBE.P11 TST62

RH11-RS03LA-RS03-RS04 BASIC FUNCTION DIAGNOSTIC MACY11 27(732) 25-SEP-76 09:06 PAGE 58
TEST ILLEGAL FUNCTION CODE 67

```

2229 ;*****
2230 ;TEST 62 TEST ILLEGAL FUNCTION CODE 67
2231 ;*****
2232 011332 104400 TST62: SCOPE
2233
2234 011334 104414 ILF67: CLRDK ;CLEAR ALL RS REG
2235 011336 012777 177777 167544 MOV #-1, JRSWC ;SET WC TO -1
2236 011344 013737 001102 001174 MOV J#0BUFSV, WORK ;GET OUTBUF ADD.
2237 011352 062737 000002 001174 ADD #2, WORK ;FOR TEST
2238 011360 013777 001174 167524 MOV WORK, JRSBA ;LOAD ADDR.
2239 011366 012777 000067 167510 MOV #67, JRS1 ;DO FUNCTION 67
2240 011374 105777 167504 15: TSTB JRS1 ;DONE YET?
2241 011400 100375 BPL 15 ;NO
2242 011402 017700 167504 MOV JRSBA, BAD ;GET BA REG
2243 011406 013701 001102 MOV J#0BUFSV, GOOD ;GET CORRECT ANS FOR RSBA
2244 011412 020100 CMP GOOD, BAD ;IS RSBA CORRECT?
2245 011414 001401 BEQ .+4 ;YES
2246 011416 104000 HLT ;BAD=RSBA GOOD=CORRECT ANS.
2247 011420 013701 001152 MOV UNNUM, GOOD ;GET UNIT NUMBER
2248 011424 052701 001300 BIS #1300, GOOD ;SET IR AND OR BITS
2249 011430 017700 167452 MOV JRS1, BAD ;GET CS2
2250 011434 020100 CMP GOOD, BAD ;IS CS2 CORRECT?
2251 011436 001401 BEQ .+4 ;YES
2252 011440 104000 HLT ;BAD=CS2 GOOD=CORRECT ANS
2253 011442 022777 000001 167450 CMP #1, JRSER ;IS RSER CORRECT?
2254 011450 001401 BEQ .+4 ;YES
2255 011452 104002 HLT ;ER IS WRONG
2256 011454 104414 CLRDK ;CLEAR ALL RS REG
2257 011456 005777 167436 TST JRSER ;DID ERROR CLEAR
2258 011462 001401 BEQ .+4 ;YES
2259 011464 104040 HLT ;NO
2260 011466 012777 177700 167414 MOV #-100, JRSWC ;SET WC TO -100
2261 011474 013737 001102 001174 MOV J#0BUFSV, WORK ;GET OUTBUF ADD.
2262 011502 062737 000200 001174 ADD #200, WORK ;FOR TEST
2263 011510 013777 001174 167374 MOV WORK, JRSBA ;LOAD ADDR
2264 011516 012777 000067 167360 MOV #67, JRS1 ;DO FUNCTION 67
2265 011524 105777 167354 25: TSTB JRS1 ;DONE YET?
2266 011530 100375 BPL 25 ;NO

```



```

2267 011532 013701 001102 3$: MOV 2#0BUFSV,GOOD ;GET CORRECT ANS.
2268 011536 017700 167350 MOV 2RSBA,BAD ;GET BA REG
2269 011542 020100 CMP GOOD,BAD ;IS RSBA CORRECT?
2270 011544 001401 BEQ .+4 ;YES
2271 011546 104000 HLT ;BAD=RSBA GOOD=CORRECT ANS.
2272 011550 013701 001152 MOV UNNUM,GOOD ;GET UNIT NUMBER
2273 011554 052701 001300 BIS #1300,GOOD ;SET IR AND OR BITS
2274 011560 017700 167322 MOV 2RSCS2,BAD ;GET CS2
2275 011564 020100 CMP GOOD,BAD ;IS CS2 CORRECT?
2276 011566 001401 BEQ .+4 ;YES
2277 011570 104000 HLT ;BAD=CS2 GOOD=CORRECT ANS
2278 011572 022777 000001 167320 CMP #1,2RSER ;IS RSER CORRECT?
2279 011600 001401 BEQ .+4 ;YES
2280 011602 104002 HLT ;ER IS WRONG
2281 011604 012777 040011 167272 MOV #40011,2RSCS1 ;CLEAR ERRORS
2282 011612 022777 004210 167264 CMP #4210,2RSCS1 ;DID THEY CLEAR IN CS1
2283 011620 001401 BEQ .+4 ;YES
2284 011622 104040 HLT ;NO
2285 011624 005777 167270 TST 2RSER ;DID RSER CLEAR
2286 011630 001401 BEQ .+4 ;YES
2287 011632 104040 HLT ;NO
2288 011634 005777 167262 TST 2RSAS ;DID RSAS CLEAR
2289 011640 001401 BEQ .+4 ;YES
2290 011642 104100 HLT ;NO

```

```

*****
;TEST 63 TEST PAR IN RSER
*****
TST63: SCOPE

```

```

2295 011644 104400
2296
2297 011646 013737 001164 022414 PARTST: MOV TIMSV,TIMES ;RESTORE LOOP #
2298 011654 104414 CLRDK ;CLEAR ALL RS REG
2299 011656 012777 000010 167234 1$: MOV #10,2RSER ;SET PAR
2300 011664 022777 150600 167224 CMP #150600,2RSDS ;DID ERR,ATA AND DRY SET?
2301 011672 001401 BEQ .+4 ;YES
2302 011674 104042 HLT ;ER SHOULD SET IF PAR SETS IN RSER
2303 011676 104414 CLRDK ;CLEAR ALL RS REG
2304 011700 005777 167214 TST 2RSER ;DID PAR CLEAR?
2305 011704 001401 BEQ .+4 ;YES
2306 011706 104002 HLT ;ER ;PAR DID NOT CLEAR BY CLEAR BIT
2307 011710 022777 010600 167200 CMP #10600,2RSDS ;DID ERROR BITS CLEAR
2308 011716 001401 BEQ .+4 ;YES
2309 011720 104040 HLT ;NO

```


H05

MAINDEC-11-DZRSB-E
DZRSBE.P11 TST63RH11-RS03LA-RS03-RS04 BASIC FUNCTION DIAGNOSTIC MACY11 27(732) 25-SEP-76 09:06 PAGE 60
TEST PAR IN RSER

```

2310 ;CHECK BITS 12 TO 15 FOR 0
2311 ;CHECK SECTOR FRACTION TO WATCH FOR MOVEMENT
2312 ;CHECK CS BITS IN LA AND ADDRESS CONFIRM IN MR REG
2313
2314 ;*****
2315 ;TEST 64 LOOK AHEAD TEST
2316 ;*****
2317 011722 104400 TST64: SCOPE
2318
2319 011724 032777 170000 167172 LATST: BIT #170000,RSLA ;ARE BITS 12 TO 15 CLEARED?
2320 011732 001401 BEQ +4 ;YES
2321 011734 104204 HLT !LA ;BITS 12 TO 15 SHOULD BE CLEARED
2322 011736 104400 TST65: SCOPE
2323
2324 ;NOW TEST MOVEMENT IN SF BITS
2325
2326 011740 012737 171005 001174 MOV #171005,WORK ;SET UP FOR TIME OUT ROUTINE
2327 011746 017701 167152 MOV RSRLA,GOOD ;GET READING FROM LA
2328 011752 042701 007700 BIC #7700,GOOD ;GET RID OF CS BITS
2329 011756 005337 001174 1$: DEC WORK ;WAIT FOR DISK
2330 011762 001407 BEQ ERRR ;TYPE ERROR
2331 011764 017700 167134 MOV RSRLA,BAD ;READ LA
2332 011770 042700 007700 BIC #7700,BAD ;CLEAR CS BITS
2333 011774 020100 CMP GOOD,BAD ;DID SF BITS CHANGE?
2334 011776 001767 BEQ 1$ ;WAIT FOR TIME OUT
2335 012000 000422 BR LATDON ;LA OK CONT
2336 012002
2337 012002 104402 012006 ERRR: TYPE +2 ;.ASCIZ <15><12>"SECTOR FRACTIONS NOT MOVING"
2338 012044 104204 HLT !LA ;TYPE LOOK AHEAD REG
2339 012046 LATDON: ;DONE CONT.

```



```

2340 ;*****
2341 ;TEST 66 CHECK CS BITS TO INCREMENT AND ADDRESS CONFIRM BIT IN MR
2342 ;*****
2343 012046 104400 TST66: SCOPE
2344
2345 012050 013737 022414 001164 CSTST: MOV TIMES,TIMSV ;SAVE LOOP CT
2346 012056 012737 000010 022414 MOV #10,TIMES ;LOOP 10 TIMES
2347 012064 104414 CLRDK ;CLEAR ALL RS REG.
2348 012066 012701 001000 MOV #1000,GOOD ;LOAD COUNTER
2349 012072 032777 001000 167030 BIT #BIT9,ARSMR ;IS ADD CONFIRM BIT 0?
2350 012100 001407 BEQ ADDCF ;YES CONTINUE
2351 012102 005301 DEC GOOD ;WAIT FOR
2352 012104 001376 BNE -2 ;DISK TO MOVE
2353 012106 032777 001000 167014 BIT #BIT9,ARSMR ;IS ADD. CON. BIT BIT 0?
2354 012114 001401 BEQ +4 ;YES
2355 012116 104220 HLT !MR ;ADD. CONF. BIT ALWAYS A 1
2356
2357 ;NOW TEST TA BITS AND ADD. CON. BIT IN MR
2358
2359 012120 012777 177777 166766 ADDCF: MOV #-1,ARSDA ;INIT RSDA
2360 012126 012737 177777 001174 1$: MOV #-1,WORK ;SETUP TIMEOUT COUNTER
2361 012134 005277 166754 INC ARSDA ;GET NEXT SECTOR
2362 012140 022777 010000 166746 CMP #10000,ARSDA ;DONE ALL YET?
2363 012146 001433 BEQ DONCS ;YES
2364 012150 005337 001174 2$: DEC WORK ;DO TIMEOUT ROUTINE
2365 012154 001427 BEQ TMEOUT ;ADD. CON. NEVER CAME UP
2366 012156 032777 001000 166744 BIT #1000,ARSMR ;DID ADD CONFIRM BIT SET?
2367 012164 001771 BEQ 2$ ;YES
2368 012166 017700 166732 MOV ARSLA,BAD ;GET LA
2369 012172 042700 000077 BIC #77,BAD ;CLEAR SF BITS
2370 012176 012737 000006 001176 MOV #6,WORK1 ;SET UP COUNTER
2371 012204 006000 3$: ROR BAD ;MOV SA BITS RIGHT
2372 012206 005337 001176 DEC WORK1 ;DO 6 TIMES
2373 012212 001374 BNE 3$ ;DONE YET?
2374 012214 017701 166674 MOV ARSDA,GOOD ;GET DA
2375 012220 042701 177700 BIC #177700,GOOD ;CLEAR JUNK
2376 012224 020100 CMP GOOD,BAD ;ARE SA BITS = IN DA AND LA REG.?
2377 012226 001401 BEQ +4 ;OK
2378 012230 104000 HLT ;GOOD =DA BAD = LA
2379 012232 000735 BR 1$ ;NO WAIT
2380
2381 012234 104262 TMEOUT: HLT !MR!ER!DS ;ADDRESS CONFIRM BIT NEVER SET COULD BE BAD OR
2382 ;BAD LA OR BAD COMPARE BETWEEN LA AND DA
2383 012236 DONCS: ;TEST DONE CONTINUE

```



```

2384 ;*****
2385 ;TEST 67 PARITY TEST
2386 ;*****
2387 012236 104400 TST67: SCOPE
2388
2389 012240 013737 001164 022414 PART: MOV TIMSV,TIMES ;RESTORE LOOP COUNTER
2390 012246 012737 012426 000004 MOV #PRT,2#4 ;SETUP TIME OUT VECTOR
2391 012254 012737 000340 000006 MOV #340,2#6
2392 012262 012702 172100 MOV #MPRO,R2 ;GET PAR REG
2393 012266 005712 TSTAGN: TST (R2) ;DOES IT EXIST
2394 012270 012712 000004 MOV #WMP,2R2 ;YES SET WRITE WRONG PARITY
2395 012274 012701 001174 MOV #WORK,R1 ;GET TEST LOCATION
2396 012300 011111 1S: MOV 2R1,2R1 ;WRITE WRONG PARITY
2397 012302 005711 TST 2R1 ;READ IT
2398 012304 005712 TST 2R2 ;DID PARITY ERROR SET?
2399 012306 100402 BMI 2S ;YES
2400 012310 005012 CLR 2R2 ;CLEAR PARITY REG
2401 012312 000446 BR PRT1 ;GET NEXT PARITY REG
2402 012314 012712 000001 2S: MOV #1,2R2 ;CLEAR WMP AND ENABLE BAD PARITY
2403 012320 104414 CLRDK ;CLEAR ALL RS REG
2404 012322 012737 177777 027364 MOV #177777,OUTBUF ;DATA TO BE X-FERED
2405 012330 012777 001174 166554 MOV #WORK,2RSBA ;SET UP CURRENT ADDRESS
2406 012336 012777 177777 166544 MOV #-1,2RSCW ;SET WORD COUNT TO -1
2407 012344 012777 000061 166532 MOV #61,2RSCS1 ;GO WRITE
2408 012352 105777 166526 3S: TSTB 2RSCS1 ;DONE YET?
2409 012356 100375 BPL 3S ;NO WAIT
2410 012360 005012 CLR 2R2 ;CLEAR PARITY REG
2411 012362 005037 001174 CLR WORK ;WRITE GOOD PARITY
2412 012366 017700 166514 MOV 2RSCS2,BAD ;GET CS2
2413 012372 012701 020100 MOV #20100,GOOD
2414 012376 053701 001152 BIS UNNUM,GOOD ;GET CORRECT AND FOR CS2
2415 012402 020100 CMP GOOD,BAD ;IS CS2 CORRECT?
2416 012404 001401 BEQ .+4 ;YES
2417 012406 104000 HLT ;CS2 SHOULD = GOOD
2418 012410 022777 144260 166466 CMP #144260,2RSCS1 ;IS CS1 CORRECT?
2419 012416 001401 BEQ .+4 ;YES
2420 012420 104040 HLT ;DS
2421 012422 000137 012446 JMP NOPAR ;GET OUT
2422
2423 ;TRAPOUT ROUTINE
2424
2425 012426 022626 PRT1: CMP (6)+(6)+ ;CLEAR STACK
2426 012430 022702 172136 PRT1: CMP #172136,R2 ;DONE YET?
2427 012434 001404 BEQ NOPAR ;YES NO PAR REG
2428 012436 062702 000002 ADD #2,R2 ;NO TRY AGAIN
2429 012442 000137 012266 JMP TSTAGN ;RETRY
2430
2431 012446 012737 000006 000004 NOPAR: MOV #6,2#4
2432 012454 005037 000006 CLR 2#6

```


K05

 MAINDEC-11-DZRSB-E
 DZRSBE.P11 TST70

 RH11-RS03LA-RS03-RS04 BASIC FUNCTION DIAGNOSTIC MACY11 27(732) 25-SEP-76 09:06 PAGE 63
 TEST WRITE CHECK ERROR

```

2433      ;*****
2434      ;TEST 70          TEST WRITE CHECK ERROR
2435      ;*****
2436      012460 104400
2437      TST70: SCOPE
2438      ;WRITE A WORD OF 0 AND FLOAT A 1 THROUGH IT TO CAUSE WCE
2439      ;SET BIT14 IN ONCE AND WRITE A WD OF -1 AND FLOAT 0
2440      ;TO CAUSE WCE
2441
2442      012462 104414      WCETST: CLRDK      ;CLEAR ALL RS REG
2443      012464 005037 027364      CLR      OUTBUF      ;WRITE A WD OF 0
2444      012470 013777 001102 166414      WCE TT: MOV      @#0BUFSV, @RSBA      ;SET UP CURRENT ADDRESS
2445      012476 012777 177777 166404      MOV      #-1, @RSWC      ;SET WORD COUNT TO -1
2446      012504 012777 000061 166372      MOV      #61, @RSCS1      ;GO WRITE
2447      012512 105777 166366      3$:      TSTB      @RSCS1      ;DONE YET?
2448      012516 100375      BPL      3$      ;NO WAIT
2449      012520 032737 040000 001160      BIT      #BIT14, ONCE      ;WRITE A 1 OR 0?
2450      012526 001410      BEQ      2$      ;WRITE A 0
2451      012530 012737 177777 027364      MOV      #-1, OUTBUF      ;WRITE A 1
2452      012536 000241      CLC      ;CLEAR CARRY
2453      012540 006137 027364      6$:      ROL      OUTBUF      ;FLOAT A 0 THROUGH BAD WD
2454      012544 103123      BCC      WCEDON      ;DONE GET OUT
2455      012546 000406      BR      5$      ;CHECK WCE
2456      012550 005037 027364      2$:      CLR      OUTBUF      ;WRITE A 0
2457      012554 000261      SEC      ;SET CARRY
2458      012556 006137 027364      1$:      ROL      OUTBUF      ;FLOAT A 1
2459      012562 103503      BCS      WCEDNE      ;GET OUT WHEN DONE
2460      012564 013777 001102 166320      5$:      MOV      @#0BUFSV, @RSBA      ;SET UP CURRENT ADDRESS
2461      012572 012777 177777 166310      MOV      #-1, @RSWC      ;SET WORD COUNT TO -1
2462      012600 005077 166310      CLR      @RSDA
2463      012604 012777 000051 166272      MOV      #51, @RSCS1      ;GO WRITE CHECK
2464      012612 105777 166266      4$:      TSTB      @RSCS1      ;READY YET?
2465      012616 100375      BPL      4$      ;NO WAIT
2466      012620 017700 166262      MOV      @RSCS2, BAD      ;GET CS2
2467      012624 013701 001152      MOV      UNNUM, GOOD      ;SET UNIT #
2468      012630 052701 040300      BIS      #40300, GOOD      ;SET BITS
2469      012634 020100      CMP      GOOD, BAD      ;IS CS2 CORRECT?
2470      012636 001413      BEQ      7$      ;YES
2471      012640 104000      HLT
2472      012642 013700 027364      MOV      OUTBUF, BAD      ;BAD=CS2 GOOD=CORRECT ANS
2473      012646 005001      CLR      GOOD      ;GET BAD WD THAT SHOULD CAUSE WCE
2474      012650 032737 040000 001160      BIT      #BIT14, ONCE      ;GET GOOD WD IF WRITING 0
2475      012656 001402      BEQ      8$      ;ARE WE WRITING 1 OR 0
2476      012660 012701 177777      MOV      #-1, GOOD      ;0
2477      012664 104000      8$:      HLT      ;GET GOOD WD FOR 1
2478      ;GOOD = CORRECT WD WRITTEN
2479      ;BAD = INCORRECT WD THAT WCE DID NOT CATCH

```


MAINDEC-11-DZRSB-E
DZRSBE.P11

TST70

RH11-R503LA-R503-R504 BASIC FUNCTION DIAGNOSTIC MACY11 27(732) 25-SEP-76 09:06 PAGE 64
TEST WRITE CHECK ERROR

2479	012666	022777	144250	166210	7\$:	CMP	#144250, @RSCS1	;DID TRE SET?
2480	012674	001401				BEQ	.+4	;YES
2481	012676	104043				HLT	!CS1!ER!DS	;TRE SHOULD SET IF WCE SETS
2482	012700	017700	166206			MOV	@RSBA, BAD	;FETCH CURRENT ADDRESS
2483	012704	013701	001102			MOV	@#OBUFSV, GOOD	;WHAT RSBA SHOULD EQUAL
2484	012710	062701	000002			ADD	#2, GOOD	;UPDATE IT
2485	012714	020001				CMP	BAD, GOOD	;IS RSBA CORRECT
2486	012716	001401				BEQ	.+4	;YES EXECUTE CONTINUE
2487	012720	104000				HLT		;RSBA FAILED TO INCREMENT
2488	012722	104414				CLRDK		;CLEAR ALL RS REG
2489	012724	013701	001152			MOV	UNNUM, GOOD	;PUT DRIVE IN GOOD
2490	012730	052701	000100			BIS	#100, GOOD	;SET IR BIT
2491	012734	017700	166146			MOV	@RSCS2, BAD	;GET CS2
2492	012740	020100				CMP	GOOD, BAD	;IS CS2 CORRECT
2493	012742	001401				BEQ	.+4	;YES
2494	012744	104000				HLT		;BAD =CS2 GOOD IS CORRECT ANS
2495	012746	022777	004200	166130		CMP	#4200, @RSCS1	;DID TRE CLEAR?
2496	012754	001401				BEQ	.+4	;YES
2497	012756	104001				HLT	!CS1	;TRE DID NOT CLEAR WITH CLEAR
2498	012760	032737	040000	001160		BIT	#BIT14, ONCEE	;FLOATION A 1 OR 0?
2499	012766	001673				BEQ	1\$	;FLOAT 1
2500	012770	000663				BR	6\$	;FLOAT 0
2501	012772	052737	040000	001160	WCEDNE:	BIS	#BIT14, ONCEE	;SET BIT14
2502	013000	104414				CLRDK		
2503	013002	012737	177777	027364		MOV	#-1, OUTBUF	
2504	013010	000137	012470			JMP	WCETT	;NOW WRITE -1 IN OUTBUF
2505	013014	042737	040000	001160	WCEDON:	BIC	#BIT14, ONCEE	;CLEAR TEST FLAG

M05

MAINDEC-11-DZRSB-E
DZRSBE.P11 TST71

RH11-RS03LA-RS03-RS04 BASIC FUNCTION DIAGNOSTIC MACY11 27(732) 25-SEP-76 09:06 PAGE 65
TEST WRITE CHECK ERROR ON -B- PORT

```

2506 ;*****
2507 ;TEST 71 TEST WRITE CHECK ERROR ON -B- PORT
2508 ;*****
2509 013022 104400 TST71: SCOPE
2510
2511 ;WRITE A WORD OF 0 AND FLOAT A 1 THROUGH IT TO CAUSE WCE
2512 ;SET BIT14 IN ONCEE AND WRITE A WD OF -1 AND FLOAT 0
2513 ;TO CAUSE WCE
2514
2515 013024 032737 020000 001160 WCETSB: BIT #BIT13,ONCEE ; -B- PORT?
2516 013032 001402 BEQ 1$ ; YES
2517 013034 000137 013400 JMP WCEDOS ; NO GET OUT
2518 013040 013737 001170 001102 1$: MOV BPORTT, OBUFSV ; GET -B- PORT BUFFER
2519 013046 104414 CLRDK ; CLEAR ALL RS REG
2520 013050 005077 166114 CLR @BPORTT ; WRITE A WD OF 0
2521 013054 013777 001102 166030 WCETB: MOV @#OBUFSV, @RSBA ; SET UP CURRENT ADDRESS
2522 013062 012777 177777 166020 MOV #-1, @RSWC ; SET WORD COUNT TO -1
2523 013070 012777 002061 166006 MOV #2061, @RSCS1 ; GO WRITE
2524 013076 105777 166002 3$: TSTB @RSCS1 ; DONE YET?
2525 013102 100375 BPL 3$ ; NO WAIT
2526 013104 032737 040000 001160 BIT #BIT14,ONCEE ; WRITE A 1 OR 0?
2527 013112 001410 BEQ 2$ ; WRITE A 0
2528 013114 012777 177777 166046 MOV #-1, @BPORTT ; WRITE A 1
2529 013122 000241 CLC ; CLEAR CARRY
2530 013124 006137 027364 6$: ROL OUTBUF ; FLOAT A 0 THROUGH BAD WD
2531 013130 103123 BCC WCEDOS ; DONE GET OUT
2532 013132 000406 BR 5$ ; CHECK WCE
2533 013134 005077 166030 2$: CLR @BPORTT ; WRITE A 0
2534 013140 000261 SEC ; SET CARRY
2535 013142 006177 166022 1$: ROL @BPORTT ; FLOAT A 1
2536 013146 103503 BCS WCEDNB ; GET OUT WHEN DONE
2537 013150 013777 001102 165734 5$: MOV @#OBUFSV, @RSBA ; SET UP CURRENT ADDRESS
2538 013156 012777 177777 165724 MOV #-1, @RSWC ; SET WORD COUNT TO -1
2539 013164 005077 165724 CLR @RSDA
2540 013170 012777 002051 165706 4$: MOV #2051, @RSCS1 ; GO WRITE CHECK
2541 013176 105777 165702 TSTB @RSCS1 ; READY YET?
2542 013202 100375 BPL 4$ ; NO WAIT
2543 013204 017700 165676 MOV @RSCS2, BAD ; GET CS2
2544 013210 013701 001152 MOV UNNUM, GOOD ; SET UNIT #
2545 013214 052701 040300 BIS #40300, GOOD ; SET BITS
2546 013220 020100 CMP GOOD, BAD ; IS CS2 CORRECT?
2547 013222 001413 BEQ 7$ ; YES
2548 013224 104000 HLT ; CS2=BAD GOOD=CORRECT ANS
2549 013226 017700 165736 MOV @BPORTT, BAD ; GET BAD WD THAT SHOULD CAUSE WCE
2550 013232 005001 CLR GOOD ; GET GOOD WD IF WRITING 0
2551 013234 032737 040000 001160 BIT #BIT14,ONCEE ; ARE WE WRITING 1 OR 0
2552 013242 001402 BEQ 8$ ; 0
2553 013244 012701 177777 8$: MOV #-1, GOOD ; GET GOOD WD FOR 1
2554 013250 104000 HLT ; GOOD = CORRECT WD WRITTEN
2555 ; BAD = INCORRECT WD THAT WCE DID NOT CATCH

```


N05

MAINDEC-11-DZRSB-E
DZRSBE.P11 TST71

RH11-RS03LA-RS03-RS04 BASIC FUNCTION DIAGNOSTIC MACY11 27(732) 25-SEP-76 09:06 PAGE 66
TEST WRITE CHECK ERROR ON -B- PORT

2556	013252	022777	146250	165624	7\$:	CMP	#146250, @RSCS1	;DID TRE SET?
2557	013260	001401				BEQ	.+4	;YES
2558	013262	104043				HLT	!CS1!ER!DS	;TRE SHOULD SET IF WCE SETS
2559	013264	017700	165622			MOV	@RSBA, BAD	;FETCH CURRENT ADDRESS
2560	013270	013701	001102			MOV	@#OBUFSV, GOOD	;WHAT RSBA SHOULD EQUAL
2561	013274	062701	000002			ADD	#2, GOOD	;UPDATE IT
2562	013300	020001				CMP	BAD, GOOD	;IS RSBA CORRECT
2563	013302	001401				BEQ	.+4	;YES EXECUTE CONTINUE
2564	013304	104000				HLT		;RSBA FAILED TO INCREMENT
2565	013306	104414				CLRDK		;CLEAR ALL RS REG
2566	013310	013701	001152			MOV	UNNUM, GOOD	;PUT DRIVE IN GOOD
2567	013314	052701	000100			BIS	#100, GOOD	;SET IR BIT
2568	013320	017700	165562			MOV	@RSCS2, BAD	;GET CS2
2569	013324	020100				CMP	GOOD, BAD	;IS CS2 CORRECT
2570	013326	001401				BEQ	.+4	;YES
2571	013330	104000				HLT		;BAD =CS2 GOOD IS CORRECT ANS
2572	013332	022777	004200	165544		CMP	#4200, @RSCS1	;DID TRE CLEAR?
2573	013340	001401				BEQ	.+4	;YES
2574	013342	104001				HLT	!CS1	;TRE DID NOT CLEAR WITH CLEAR
2575	013344	032737	040000	001160		BIT	#BIT14, ONCEE	;FLOATION A 1 OR 0?
2576	013352	001673				BEQ	1\$	;FLOAT 1
2577	013354	000663				BR	6\$	;FLOAT 0
2578	013356	052737	040000	001160	WCEDNB:	BIS	#BIT14, ONCEE	;SET BIT14
2579	013364	104414				CLRDK		
2580	013366	012777	177777	165574		MOV	#-1, @BPORTT	
2581	013374	000137	013054			JMP	WCETB	;NOW WRITE -1 IN OUTBUF
2582	013400	012737	027364	001102	WCEDOS:	MOV	#OUTBUF, OBUFSV	;RESTORE OBUFSV
2583	013406	042737	040000	001160		BIC	#BIT14, ONCEE	;CLEAR TEST FLAG


```

2584
2585
2586
2587 013414 104400
2588
2589 013416 104414
2590 013420 012737 177777 027364
2591 013426 013777 001102 165456
2592 013434 012777 177000 165446
2593 013442 012777 000061 165434
2594 013450 105777 165430
2595 013454 100775
2596 013456 012777 000071 165420
2597 013464 004737 026600
2598 013470 104001
2599 013472 022777 144260 165404
2600 013500 001401
2601 013502 104001
2602 013504 005777 165412
2603 013510 001401
2604 013512 104100
2605 013514 013701 001152
2606 013520 052701 002300
2607 013524 017700 165356
2608 013530 020100
2609 013532 001401
2610 013534 104000
2611 013536 005777 165346
2612 013542 001001
2613 013544 104011
2614 013546 005777 165346
2615 013552 001401
2616 013554 104040
2617 013556 052777 040000 165320
2618 013564 042701 002000
2619 013570 017700 165312
2620 013574 020100
2621 013576 001401
2622 013600 104200
2623 013602 022777 004260 165274
2624 013610 001401
2625 013612 104040

```

```

*****
:TEST 72 TEST PROGRAM ERROR BIT IN RSCS2
*****

```

TST72: SCOPE

PGETST: CLRDK

```

MOV #177777,OUTBUF ;CLEAR ALL RS REG
MOV @#0BUFSV,@RSBA ;DATA TO BE X-FERED
MOV #177000,@RSMC ;SET UP CURRENT ADDRESS
MOV #61,@RSCS1 ;SET WORD COUNT
TSTB @RSCS1 ;GO WRITE
BMT 25 ;IS RDY CLEARED YET?
MOV #71,@RSCS1 ;NO WAIT
JSR PC,WAITRY ;GO READ
HLT !CS1 ;WAIT FOR READY
CMP #144260,@RSCS1 ;RDY NEVER CAME UP
BEQ .+4 ;IS CS1 CORRECT?
HLT !CS1 ;YES
TST @RSAS ;TRE SHOULD SET BY SETTING PGE
BEQ .+4 ;AS SHOULD = 0
HLT !AS ;YES
MOV UNNUM,GOOD ;RSAS SHOULD = 0
BIS #2300,GOOD ;GET UNIT #
MOV @RSCS2,BAD ;SET PGE, IR, AND OR
CMP GOOD,BAD ;GET CS2
BEQ .+4 ;IS IT CORRECT?
HLT !CS1 ;YES
TST @RSMC ;BAD = CS2
BNE .+4 ;SHOULD NOT BE 0
HLT !WC!CS1 ;BECAUSE PGE SHOULD ABORT
TST @RSER ;CURRENT OPERATION
BEQ .+4 ;DID ANY ERRORS SET?
HLT !DS ;NO
BIS #TRE,@RSCS1 ;RMR SHOULD BE SET
BIC #PGE,GOOD ;CLEAR ERRORS
MOV @RSCS2,BAD ;CLEAR PGE ERROR
CMP GOOD,BAD ;GET CS2
BEQ .+4 ;IS CS2 CORRECT?
HLT !CS2 ;YES
CMP #4260,@RSCS1 ;PGE DID NOT CLEAR BY CLEARING TRE BAD = CS2
BEQ .+4 ;DID SC CLEAR
HLT !DS ;YES
;DID NOT CLEAR BY CLEARING TRE

```



```

2626                                     ;*****
2627                                     ;TEST 73                                TEST RMR IN RSER REGISTER TRYING TO WRITE INTO RSDA
2628                                     ;*****
2629 013614 104400                         TST73: SCOPE
2630
2631 013616 104414                         RMRT1: CLRDK
2632 013620 013777 001102 165264          MOV      @#0BUFSV, @RSBA      ;CLEAR ALL RS REG
2633 013626 012777 177700 165254          MOV      @177700, @RSMC    ;SET UP CURRENT ADDRESS
2634 013634 012703 172060                  MOV      @172060, R3      ;SET WORD COUNT
2635 013640 011304                          15:  MOV      @R3, R4      ;GET RSLA REG
2636 013642 042704 000017                  BIC      @17, R4      ;WAIT FOR
2637 013646 022704 000020                  CMP      @20, R4      ;THE MIDDLE
2638 013652 001372                          BNE      15           ;OF SECTOR 0
2639 013654 012777 000031 165222          MOV      @31, @RSCS1    ;BEFORE DOING A SEARCH
2640 013662 012777 007777 165224          MOV      @7777, @RSDA  ;SEARCH
2641 013670 004737 026600                  JSR      PC, WAITRY    ;CAUSE ERROR
2642 013674 104001                          HLT      !CS1          ;WAIT FOR READY
2643 013676 022777 000004 165214          CMP      @4, @RSER     ;RDY NEVER CAME UP
2644 013704 001401                          BEQ      , +4          ;DID RMR SET?
2645 013706 104002                          HLT      !ER           ;YES
2646 013710 022777 007777 165176          CMP      @7777, @RSDA  ;ER SHOULD = 4
2647 013716 001001                          BNE      , +4          ;DID DA GET MODIFIED?
2648 013720 104004                          HLT      !DA           ;NO
2649 013722 022777 150600 165166          CMP      @150600, @RSDS ;DID ERR SET?
2650 013730 001401                          BEQ      , +4          ;YES
2651 013732 104042                          HLT      !DS!ER        ;ER DID NOT SET BECAUSE OF RMR
2652 013734 022777 104230 165142          CMP      @104230, @RSCS1 ;IS CS1 CORRECT?
2653 013742 001401                          BEQ      , +4          ;YES
2654 013744 104040                          HLT      !DS           ;CS1 SHOULD = 144260
2655 013746 104414                          CLRDK
2656 013750 005777 165144                  TST      @RSER         ;CLEAR ALL RS REG
2657 013754 001401                          BEQ      , +4          ;DID RMR CLEAR?
2658 013756 104002                          HLT      !ER           ;YES
2659 013760 022777 004200 165116          CMP      @4200, @RSCS1  ;RMR DID NOT CLEAR WITH A CLEAR
2660 013766 001401                          BEQ      , +4          ;IS CS1 CORRECT?
2661 013770 104040                          HLT      !DS           ;YES
2662                                     ;NO
2663                                     ;*****
2664                                     ;TEST 74                                TEST RMR IN RSER REGISTER TRYING TO WRITE INTO RSER
2665                                     ;*****
2666 013772 104400                         TST74: SCOPE
2667
2668 013774 104414                         RMRT2: CLRDK
2669 013776 013777 001102 165106          MOV      @#0BUFSV, @RSBA ;CLEAR ALL RS REG
2670 014004 012777 177700 165076          MOV      @177700, @RSMC ;SET UP CURRENT ADDRESS
2671 014012 012777 000061 165064          MOV      @61, @RSCS1    ;SET WORD COUNT
2672 014020 105777 165060                  25:  TSTB      @RSCS1    ;GO WRITE
2673 014024 100775                          BMI      25           ;IS RDY SET?
2674 014026 012777 177773 165064          MOV      @177773, @RSER ;YES WAIT FOR IT TO CLEAR
                                     ;CAUSE ERROR

```


MAINDEC-11-DZRSB-E
DZRSBE.P11

TST74

RH11-RS03LA-RS03-RS04 BASIC FUNCTION DIAGNOSTIC MACY11 27(732) 25-SEP-76 09:06 PAGE 69
TEST RMR IN RSER REGISTER TRYING TO WRITE INTO RSER

2675	014034	004737	026600		JSR	PC WAITRY	
2676	014040	104001			HLT	!CS1	:RDY NEVER CAME UP
2677	014042	022777	000004	165050	CMP	#4,RSER	:DID RMR SET?
2678	014050	001401			BEQ	+4	:YES
2679	014052	104002			HLT	!ER	:ER SHOULD = 4
2680	014054	022777	150600	165034	CMP	#150600,RSDS	:DID ERR SET?
2681	014062	001401			BEQ	+4	:YES
2682	014064	104042			HLT	!DS!ER	:ERR DID NOT SET BECAUSE OF RMR
2683	014066	022777	144260	165010	CMP	#144260,RSRCS1	:IS CS1 CORRECT?
2684	014074	001401			BEQ	+4	:YES
2685	014076	104040			HLT	!DS	:CS1 SHOULD = 144260
2686	014100	104414			CLRDK		:CLEAR ALL RS REG

:TEST 75 TEST RMR IN RSER REGISTER TRYING TO WRITE INTO RSCS1

TST75: SCOPE

2691	014102	104400			RMRT3:	CLRDK	:CLEAR ALL RS REG
2692							
2693	014104	104414			MOV	#0BUFSV,RSBA	:SET UP CURRENT ADDRESS
2694	014106	013777	001102	164776	MOV	#177700,RSWC	:SET WORD COUNT
2695	014114	012777	177700	164766	MOV	#61,RSRCS1	:GO WRITE
2696	014122	012777	000061	164754	2\$:	TSTB	:IS RDY SET?
2697	014130	105777	164750		BMI	2\$	:YES WAIT FOR IT TO CLEAR
2698	014134	100775			MOV	#30,RSRCS1	:CAUSE ERROR
2699	014136	012777	000030	164740	JSR	PC WAITRY	:WAIT FOR READY
2700	014144	004737	026600		HLT	!CS1	:RDY NEVER CAME UP
2701	014150	104001			CMP	#4,RSER	:DID RMR SET?
2702	014152	022777	000004	164740	BEQ	+4	:YES
2703	014160	001401			HLT	!ER	:ER SHOULD = 4
2704	014162	104002			CMP	#150600,RSDS	:DID ERR SET?
2705	014164	022777	150600	164724	BEQ	+4	:YES
2706	014172	001401			HLT	!DS!ER	:ERRID NOT SET BECAUSE OF RMR
2707	014174	104042			CMP	#144260,RSRCS1	:IS CS1 CORRECT?
2708	014176	022777	144260	164700	BEQ	+4	:YES
2709	014204	001401			HLT	!DS	:CS1 SHOULD = 144260
2710	014206	104040			CLRDK		:CLEAR ALL RS REG
2711	014210	104414					

E06

MAINDEC-11-DZRSB-E
DZRSBE.P11 TST76RH11-R503LA-R503-R504 BASIC FUNCTION DIAGNOSTIC MACY11 27(732) 25-SEP-76 09:06 PAGE 70
TEST THAT RMR DOES NOT SET BY WRITTING INTO RSAS

```

2712 ;*****
2713 ;TEST 76 TEST THAT RMR DOES NOT SET BY WRITTING INTO RSAS
2714 ;*****
2715 014212 104400 TST76: SCOPE
2716
2717 014214 104414 RMRT4: CLRDK ;CLEAR ALL RS REG
2718 014216 013777 001102 164666 MOV #0BUFSV,RSBA ;SET UP CURRENT ADDRESS
2719 014224 012777 177700 164656 MOV #177700,RSWC ;SET WORD COUNT
2720 014232 012703 172060 MOV #172060,R3 ;GET RSLA REG
2721 014236 011304 1S: MOV #R3,R4 ;WAIT FOR
2722 014240 042704 000017 BIC #17,R4 ;THE MIDDLE
2723 014244 022704 000020 CMP #20,R4 ;OF SECTOR 0
2724 014250 001372 BNE 1S ;BEFORE DOING A SEARCH
2725 014252 012777 000031 164624 MOV #31,RSRCS1 ;SEARCH
2726 014260 012777 000000 164634 MOV #0,RSAS ;TRY TO CAUSE ERROR
2727 014266 005037 001174 CLR WORK ;CLEAR COUNTER
2728 014272 032777 000200 164616 2S: BIT #BIT7,RSRDS ;WAIT FOR DRY
2729 014300 001004 BNE 3S ;READY CONT
2730 014302 005237 001174 INC WORK ;COUNT
2731 014306 001371 BNE 2S ;RETRY
2732 014310 104001 HLT !CS1 ;RDY NEVER CAME UP
2733 014312 005777 164602 3S: TST RSRER ;DID RMR SET?
2734 014316 001401 BEQ +4 ;NO
2735 014320 104002 HLT !ER ;ER SHOULD = 0
2736 014322 022777 110600 164566 CMP #110600,RSRDS ;DID ERR SET?
2737 014330 001401 BEQ +4 ;NO
2738 014332 104042 HLT !DS!ER ;DS SHOULD = 110600
2739 014334 022777 104230 164542 CMP #104230,RSRCS1 ;IS CS1 CORRECT?
2740 014342 001401 BEQ +4 ;YES
2741 014344 104040 HLT !DS ;CS1 SHOULD = 144260
2742 014346 104414 CLRDK ;CLEAR ALL RS REG
2743 014350 022777 004200 164526 CMP #4200,RSRCS1 ;IS CS1 CORRECT?
2744 014356 001401 BEQ +4 ;YES
2745 014360 104040 HLT !DS ;NO

```



```

2746 ;*****
2747 ;TEST 77 TEST DCK IN RSER
2748 ;*****
2749 014362 104400 TST77: SCOPE
2750
2751 ;DO A WRITE AND THEN A CLEAR FUNCTION THAT SHOULD CAUSE DCK TO SET
2752
2753 DCKTST: CLRDK ;CLEAR ALL RS REG
2754 014364 104414 000004 001162 CMP #4,RS04DT ;IS THIS A LA DISK?
2755 014366 022737 000004 001162 BNE 7$ ;NO
2756 014374 001004 177640 001200 MOV #177640,WORK2 ;GET WC FOR LA DISK
2757 014376 012737 177640 001200 BR 1$ ;CONTINUE
2758 014404 000411 177500 001200 7$: MOV #177500,WORK2 ;GET WC FOR RS04
2759 014406 012737 177500 001200 TST RS04DT ;IS THIS A RS04?
2760 014414 005737 001162 BNE 1$ ;YES
2761 014422 012737 177600 001200 MOV #177600,WORK2 ;NO
2762 014430 013777 001200 164452 1$: MOV WORK2,RSWC ;LOAD WC
2763 014436 012737 177777 027364 MOV #-1,OUTBUF ;WRITE -1
2764 014444 013777 001102 164440 MOV @#0BUFSV,RSBA ;SET UP CURRENT ADDRESS
2765 014452 052777 000010 164426 4$: BIS #10,RS0CS2 ;SET BAI BIT
2766 014460 012777 000061 164416 MOV #61,RS0CS1 ;GO WRITE
2767 014466 105777 164412 5$: TSTB RS0CS1 ;IS RDY SET?
2768 014472 100375 164414 6$: BPL 5$ ;WAIT FOR WRITE TO FINISH
2769 014474 005077 027364 2$: CLR RS0DA ;SET DSK ADDRESS TO 0
2770 014500 005037 001102 164400 CLR OUTBUF ;WRITE 0
2771 014504 013777 177777 164370 MOV @#0BUFSV,RSBA ;SET UP CURRENT ADDRESS
2772 014512 012777 172060 164370 MOV #-1,RSWC ;LOAD WC
2773 014520 012702 000077 164370 3$: MOV #172060,R2 ;PUT RSLA ADDR INTO R2
2774 014524 011203 000077 164370 MOV (R2),R3 ;GET LA AND WAIT FOR
2775 014526 042703 000077 164370 BIC #77,R3 ;SECTOR 40
2776 014532 022703 004000 164370 CMP #4000,R3 ;BEFORE
2777 014536 001372 000061 164336 BNE 3$ ;WRITING
2778 014540 012777 000061 164336 6$: MOV #61,RS0CS1 ;GO WRITE
2779 014546 011203 000017 164316 MOV (R2),R3 ;GET RSLA AND WAIT FOR
2780 014550 042703 000020 164316 BIC #17,R3 ;MIDDLE OF SECTOR
2781 014554 022703 000040 164316 CMP #20,R3 ;0 BEFORE EXECUTING
2782 014560 001372 000040 164316 BNE 6$ ;A CLEAR FUNCTION
2783 014562 012777 001152 164310 MOV #40,RS0CS2 ;CLEAR ALL REG. DO IT THIS WAY
2784 014570 013777 001152 164310 MOV UNNUM,RS0CS2 ;DO NOT USE TRAP

```


2785	014576	105777	164302		INCW:	TSTB	ARSCS1	: IS BUSY CLEARED
2786	014602	100401				BMI	6\$	: FLAG CLEARED
2787	014604	104001				HLT	:CS1	: RDY NEVER CAME UP
2788	014606	013777	001102	164276	6\$:	MOV	ARBUFSV, ARSBA	: SET UP CURRENT ADDRESS
2789	014614	013777	001200	164266		MOV	WORK2, ARSWC	: LOAD WC
2790	014622	012777	000071	164254		MOV	#71, ARSCS1	: GO READ
2791	014630	105777	164250		5\$:	TSTB	ARSCS1	: IS RDY SET?
2792	014634	100375				BPL	5\$	: WAIT FOR READ TO FINISH
2793	014636	022777	100000	164254		CMP	#100000, ARSER	: DID DCK SET?
2794	014644	001401				BEQ	.+4	: YES
2795	014646	104002				HLT	:ER	: DCK DID NOT SET
2796	014650	022777	150600	164240		CMP	#150600, ARSDS	: DID ERR SET?
2797	014656	001401				BEQ	.+4	: YES
2798	014660	104040				HLT	:DS	: ER DID NOT SET BY DCK
2799	014662	022777	144270	164214		CMP	#144270, ARSCS1	: IS CS1 CORRECT?
2800	014670	001401				BEQ	.+4	: YES
2801	014672	104044				HLT	:DS!DA	
2802	014674	017700	164206			MOV	ARSCS2, BAD	: GET CS2
2803	014700	013701	001152			MOV	UNNUM, GOOD	: GET UNIT #
2804	014704	052701	000100			BIS	#100, GOOD	: SET IR
2805	014710	020100				CMP	GOOD, BAD	: IS CS2 CORRECT?
2806	014712	001401				BEQ	.+4	: YES
2807	014714	104000				HLT		
2808	014716	012701	177700			MOV	#177700, GOOD	: NO
2809	014722	017700	164162		1\$:	MOV	ARSWC, BAD	: DID TRANSFER STOP AT END OF SECTOR?
2810	014726	020100				CMP	GOOD, BAD	
2811	014730	001401				BEQ	.+4	: YES
2812	014732	104000				HLT		: NO
2813	014734	012701	027364			MOV	#OUTBUF, GOOD	: GET BA
2814	014740	022737	000004	001162		CMP	#4, RS04DT	: LA DISK?
2815	014746	001003				BNE	7\$	: NO
2816	014750	062701	000100			ADD	#100, GOOD	: YES
2817	014754	000410				BR	3\$	: CONTINUE
2818	014756	005737	001162		7\$:	TST	RS04DT	: RS04?
2819	014762	001003				BNE	2\$	: YES
2820	014764	062701	000200			ADD	#200, GOOD	: RS03
2821	014770	000402				BR	3\$	
2822	014772	062701	000400		2\$:	ADD	#400, GOOD	: GET CORRECT ANS FOR BA
2823	014776	017700	164110		3\$:	MOV	ARSBA, BAD	: GET BA
2824	015002	020001				CMP	BAD, GOOD	: IS BA CORRECT?
2825	015004	001401				BEQ	.+4	: YES
2826	015006	104000				HLT		: NO
2827	015010	104414				CLDRK		: CLEAR ALL RS REG
2828	015012	005777	164102			TST	ARSER	: DID DCK CLEAR?
2829	015016	001401				BEQ	.+4	: YES
2830	015020	104002				HLT	:ER	: DCK DID NOT CLEAR WITH CLEAR
2831	015022	012777	177500	164060		MOV	#177500, ARSWC	: CLEAR DCK ON
2832	015030	013777	001102	164054		MOV	ARBUFSV, ARSBA	: DRIVE BY WRITING
2833	015036	012777	000061	164040		MOV	#61, ARSCS1	: GOOD DATA
2834	015044	105777	164034		4\$:	TSTB	ARSCS1	: ON DRIVE
2835	015050	100375				BPL	4\$	


```

2836 ;TEST THE ABILITY OF THE DISK CONTROL TO
2837 ;INCREMENT THE TRACK REGISTER.
2838
2839 ;A ONE WORD WRITE WILL BE EXECUTED
2840 ;RSDA=7777 RSWC = -1
2841 ;AT THE COMPLETION OF THE WRITE RSDA = 10000
2842 ;*****
2843 ;TEST 100 TEST DISK ADDRESS REGISTER
2844 ;*****
2845 015052 104400 TST100: SCOPE
2846
2847 015054 104414 DKADR: CLRDK ;CLEAR ALL RS REG
2848 015056 012777 177777 164024 MOV #177777,RSWC ;SET WORD COUNT TO -1
2849 015064 013777 001102 164020 MOV #0BUF5V,RSBA ;SET UP CURRENT ADDRESS
2850 015072 012777 007777 164014 MOV #7777,RSDA ;SET RSDA TO ALL ONES
2851 015100 012777 000061 163776 MOV #61,RSRCSI ;GO WRITE ONE WORD
2852 015106 004737 026600 JSR PC,WAITRY ;WAIT FOR READY
2853 015112 104001 HLT !CS1 ;RDY DID NOT COME UP
2854 015114 027727 163774 010000 5$: CMP RSDA,#10000 ;DOES RSDA=0
2855 015122 001401 BEQ +4 ;RSDA OK
2856 015124 104004 HLT !DA ;DA DID NOT INCREMENT
2857
2858 ;*****
2859 ;TEST 101 TEST IAE ERROR
2860 ;*****
2861 015126 104400 TST101: SCOPE
2862
2863 ;IAE ERROR SHOULD SET ERR,ATA AND SC BITS
2864
2865 015130 104414 IAERR: CLRDK ;CLEAR ALL RS REG
2866 015132 012777 177777 163750 MOV #177777,RSWC ;SET WC TO -1
2867 015140 013777 001102 163744 MOV #0BUF5V,RSBA ;SET UP BUS ADDRESS
2868 015146 012777 017777 163740 MOV #17777,RSDA ;SET DA TO RECEIVE ERROR
2869 015154 012777 000061 163722 MOV #61,RSRCSI ;GO WRITE ONE WD
2870 015162 105777 163716 7$: TSTB RSRCSI ;TEST FOR ERR OR RDY
2871 015166 100401 BMI +4 ;OK CONT.
2872 015170 000774 BR 7$ ;WAIT
2873 015172 022777 002000 163720 CMP #2000,RSER ;DID IAE SET?
2874 015200 001401 BEQ +4 ;YES
2875 015202 104002 HLT !ER ;IAE SHOULD BE SET
2876 015204 022777 150600 163704 CMP #150600,RSRDS ;DID ERR SET?
2877 015212 001401 BEQ +4 ;YES
2878 015214 104140 HLT !DS!AS ;ERR SHOULD BE SET
2879 015216 022777 144260 163660 CMP #144260,RSRCSI ;DID SC SET?
2880 015224 001401 BEQ +4 ;YES
2881 015226 104001 HLT !CS1 ;SC SHOULD BE SET
2882 015230 104414 CLRDK ;CLEAR ALL RS REG
2883 015232 005777 163662 TST RSRER ;CLR ERRORS?
2884 015236 001401 BEQ +4 ;YES
2885 015240 104002 HLT !ER ;ERR DID NOT CLR WITH 40 IN CS2

```



```

2886                                     ; IN THIS ROUTINE THE PROGRAM WILL GENERATE A
2887                                     ; NON-EXISTENT DISK ERROR
2888
2889                                     ; *****
2890                                     ; TEST 102                                TEST FOR NON-EXISTENT DISK ERROR
2891                                     ; *****
2892 015242 104400                         TST102: SCOPE
2893
2894 015244 104414                         NEDTST: CLRDK
2895 015246 012737 000401 001174          MOV      #401,WORK      ; CLEAR ALL RS REG
2896 015254 005001                        CLR      GOOD          ; SET UP FOR N.E.D. NUMBER
2897 015256 033737 001174 001154 1$:    BIT      WORK,UNITSV    ; LOOK FOR NON EXISTENT DRIVES
2898 015264 001405                        BEQ      3$             ; ON THE SYSTEM
2899 015266 005201                        INC      GOOD          ; FOUND NON EXISTENT DRIVE
2900 015270 006137 001174              ROL      WORK            ; CONTAINS UNIT #
2901 015274 103452                        BCS      NEDDON         ; KEEP LOOKING
2902 015276 000767                        BR       1$             ; COULD NOT FIND ANY NON EXISTENT DRIVES
2903 015300 010177 163602 3$:          MOV      GOOD,RS2CS2      ; LOOK FOR NED
2904 015304 005077 163604              CLR      RSDA           ; LOAD NED IN CS2
2905 015310 005777 163604              TST      RSER           ; WRITE DRIVE REG
2906 015314 001401                        BEQ      +4            ; DID ANY BITS SET IN RSER?
2907 015316 104040                        HLT      !DS           ; NO
2908 015320 017700 163562              MOV      RS2CS2,BAD       ; WHY DID RSER CHANGE?
2909 015324 052701 010100              BIS      #10100,GOOD      ; GET CS2
2910 015330 020100                        CMP      GOOD,BAD       ; SET NED AND IR
2911 015332 001401                        BEQ      +4            ; IS CS2 CORRECT?
2912 015334 104000                        HLT      !DS           ; YES
2913 015336 022777 160200 163540      CMP      #160200,RS2CS1    ; GOOD=CORRECT CS2 BAD=CS2
2914 015344 001401                        BEQ      +4            ; IS CS1 CORRECT?
2915 015346 104200                        HLT      !CS2          ; YES
2916 015350 005777 163546              TST      RSAS           ; TRE SHOULD SET BY NED ERROR
2917 015354 001401                        BEQ      +4            ; DID ANY BITS SET?
2918 015356 104100                        HLT      !AS           ; NO
2919 015360 112777 000100 163552      MOV      #100,RS2CS1B     ; WHY DID AT BITS SET?
2920 015366 032777 010000 163512      BIT      #NED,RS2CS2    ; CLEAR TRE
2921 015374 001401                        BEQ      +4            ; DID NED CLEAR
2922 015376 104200                        HLT      !CS2          ; YES
2923 015400 017737 163510 001174      MOV      RSDA,WORK       ; NED DID NOT CLEAR
2924 015406 032777 010000 163472      BIT      #NED,RS2CS2    ; READ DRIVE REG
2925 015414 001001                        BNE      +4            ; DID NED SET?
2926 015416 104040                        HLT      !DS           ; NED DID NOT SET
2927 015420 000431                        BR       NNDD          ; GET OUT
2928 015422 032737 010000 001160 NEDDON: BIT      #BIT12,ONCEE ; WAS THIS TYPED BEFORE?
2929 015430 001025                        BNE      NNDD          ; YES
2930 015432 104402 015436              TYPE      +2            ; ASCIZ <15><12>"COULD NOT FIND A NON-EXISTENT DRIVE"
2931 015504 052737 010000 001160 NNDD:  BIS      #BIT12,ONCEE ; SET TYPED FLAG

```



```

2932                                     ;*****
2933                                     ;TEST 103
2934                                     ;*****
2935 015512 104400                       TST103: SCOPE
2936
2937 015514 104414                       DAOTST: CLRDK
2938 015516 022737 000004 001162         CMP      #4,RS04DT
2939 015524 001004                       BNE      3$
2940 015526 012777 177737 163354         MOV      #-41,ARSWC
2941 015534 000411                       BR       1$
2942 015536 012777 177577 163344 3$:    MOV      #-201,ARSWC
2943 015544 005737 001162               TST      RS04DT
2944 015550 001003                       BNE      1$
2945 015552 012777 177677 163330         MOV      #-101,ARSWC
2946 015560 012777 007777 163326 1$:    MOV      #7777,ARSDA
2947 015566 013777 001102 163316 2$:    MOV      @#OBUFSV,ARSDA
2948 015574 012777 000061 163302         MOV      #61,ARSCS1
2949 015602 004737 026600               JSR      PC,WAITRY
2950 015606 104001                       HLT      !CS1
2951 015610 022777 001000 163302         CMP      #1000,ARSER
2952 015616 001401                       BEQ      +4
2953 015620 104002                       HLT      !ER
2954 015622 022777 152600 163266         CMP      #152600,ARSDS
2955 015630 001401                       BEQ      +4
2956 015632 104040                       HLT      !DS
2957 015634 005777 163244               TST      ARSCS1
2958 015640 100401                       BMI      +4
2959 015642 104001                       HLT      !CS1
2960 015644 104414                       CLRDK
2961 015646 022777 010600 163242         CMP      #10600,ARSDS
2962 015654 001401                       BEQ      +4
2963 015656 104040                       HLT      !DS
2964 015660 005777 163234               TST      ARSER
2965 015664 001401                       BEQ      +4
2966 015666 104002                       HLT      !ER

```

```

;CLEAR ALL RS REG
;RS03LA?
;NO
;LOAD WORD COUNT
;CONT
;LOAD WC FOR RS04
;IS THIS A RS04?
;YES
;NO
;SET RSDA=TO ALL ONES
;CURRENT ADDRESS=OUTBUF
;WRITE
;WAIT FOR READY
;RDY DID NOT SET
;DID DAO SET?
;YES
;DAO DID NOT SET
;DID LBT SET?
;YES
;LBT DID NOT SET
;IS ERROR FLAG SET
;ERROR IS SET
;SC DID NOT SET
;CLEAR ALL RS REG
;DID ATA +LBT CLEAR
;YES
;ATA DID NOT CLEAR BY CLR BIT
;DID DAO CLEAR?
;YES
;DAO DID NOT CLEAR WITH CLEAR

```


Address	Op Code	Op	Op2	Op3	Op4	Op5	Op6	Op7	Op8	Op9	Op10	Op11	Op12	Op13	Op14	Op15	Op16	Op17	Op18	Op19	Op20	Op21	Op22	Op23	Op24	Op25	Op26	Op27	Op28	Op29	Op30	Op31	Op32	Op33	Op34	Op35	Op36	Op37	Op38	Op39	Op40	Op41	Op42	Op43	Op44	Op45	Op46	Op47	Op48	Op49	Op50	Op51	Op52	Op53	Op54	Op55	Op56	Op57	Op58	Op59	Op60	Op61	Op62	Op63	Op64	Op65	Op66	Op67	Op68	Op69	Op70	Op71	Op72	Op73	Op74	Op75	Op76	Op77	Op78	Op79	Op80	Op81	Op82	Op83	Op84	Op85	Op86	Op87	Op88	Op89	Op90	Op91	Op92	Op93	Op94	Op95	Op96	Op97	Op98	Op99	Op100	Op101	Op102	Op103	Op104	Op105	Op106	Op107	Op108	Op109	Op110	Op111	Op112	Op113	Op114	Op115	Op116	Op117	Op118	Op119	Op120	Op121	Op122	Op123	Op124	Op125	Op126	Op127	Op128	Op129	Op130	Op131	Op132	Op133	Op134	Op135	Op136	Op137	Op138	Op139	Op140	Op141	Op142	Op143	Op144	Op145	Op146	Op147	Op148	Op149	Op150	Op151	Op152	Op153	Op154	Op155	Op156	Op157	Op158	Op159	Op160	Op161	Op162	Op163	Op164	Op165	Op166	Op167	Op168	Op169	Op170	Op171	Op172	Op173	Op174	Op175	Op176	Op177	Op178	Op179	Op180	Op181	Op182	Op183	Op184	Op185	Op186	Op187	Op188	Op189	Op190	Op191	Op192	Op193	Op194	Op195	Op196	Op197	Op198	Op199	Op200	Op201	Op202	Op203	Op204	Op205	Op206	Op207	Op208	Op209	Op210	Op211	Op212	Op213	Op214	Op215	Op216	Op217	Op218	Op219	Op220	Op221	Op222	Op223	Op224	Op225	Op226	Op227	Op228	Op229	Op230	Op231	Op232	Op233	Op234	Op235	Op236	Op237	Op238	Op239	Op240	Op241	Op242	Op243	Op244	Op245	Op246	Op247	Op248	Op249	Op250	Op251	Op252	Op253	Op254	Op255	Op256	Op257	Op258	Op259	Op260	Op261	Op262	Op263	Op264	Op265	Op266	Op267	Op268	Op269	Op270	Op271	Op272	Op273	Op274	Op275	Op276	Op277	Op278	Op279	Op280	Op281	Op282	Op283	Op284	Op285	Op286	Op287	Op288	Op289	Op290	Op291	Op292	Op293	Op294	Op295	Op296	Op297	Op298	Op299	Op300	Op301	Op302	Op303	Op304	Op305	Op306	Op307	Op308	Op309	Op310	Op311	Op312	Op313	Op314	Op315	Op316	Op317	Op318	Op319	Op320	Op321	Op322	Op323	Op324	Op325	Op326	Op327	Op328	Op329	Op330	Op331	Op332	Op333	Op334	Op335	Op336	Op337	Op338	Op339	Op340	Op341	Op342	Op343	Op344	Op345	Op346	Op347	Op348	Op349	Op350	Op351	Op352	Op353	Op354	Op355	Op356	Op357	Op358	Op359	Op360	Op361	Op362	Op363	Op364	Op365	Op366	Op367	Op368	Op369	Op370	Op371	Op372	Op373	Op374	Op375	Op376	Op377	Op378	Op379	Op380	Op381	Op382	Op383	Op384	Op385	Op386	Op387	Op388	Op389	Op390	Op391	Op392	Op393	Op394	Op395	Op396	Op397	Op398	Op399	Op400	Op401	Op402	Op403	Op404	Op405	Op406	Op407	Op408	Op409	Op410	Op411	Op412	Op413	Op414	Op415	Op416	Op417	Op418
---------	---------	----	-----	-----	-----	-----	-----	-----	-----	-----	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------


```

2999          ;*****
3000          ;TEST 105          EXECUTE FUNCTION WITH ERROR BITS SET
3001          ;*****
3002          TST105: SCOPE
3003
3004          016034 104400          ERTST: CLRDK          ;CLEAR ALL RS REG
3005          016036 104414          MOV          #177017, ARSER          ;LOAD ER
3006          016040 012777 177017 163052          MOV          ARSAS, BAD          ;GET AS REG
3007          016046 017700          MOV          UNCMP, GOOD          ;GET UNIT ATA BIT
3008          016052 013701 001156          BIC          #177400, GOOD          ;CLEAR JUNK
3009          016056 042701 177400          CMP          GOOD, BAD          ;IS AS REG CORRECT?
3010          016062 020100          BEQ          .+4          ;YES
3011          016066 104100          HLT          !AS          ;AS BIT SHOULD BE SET
3012          016070 022777 104200 163006          CMP          #104200, ARSCS1          ;DID ERRS SET IN CS1?
3013          016076 001401          BEQ          .+4          ;YES
3014          016100 104040          HLT          !DS          ;CS1 SHOULD =104200
3015          016102 013777 001156 163012          MOV          UNCMP, ARSAS          ;CLEAR ATA BIT
3016          016110 005777 163006          TST          ARSAS          ;DID IT CLEAR?
3017          016114 001401          BEQ          .+4          ;YES
3018          016116 104100          HLT          !AS          ;COULD NOT CLEAR AS BIT
3019          ;BY LOADING A 1 INTO IT
3020          016120 022777 004200 162756          CMP          #4200, ARSCS1          ;DID SC CLEAR BY
3021          016126 001401          BEQ          .+4          ;CLEARING ATA
3022          016130 104002          HLT          !ER          ;NO
3023          016132 012737 177777 027364          MOV          #177777, OUTBUF          ;DATA TO BE XFERED
3024          016140 013777 001102 162744          MOV          @#0BUFSV, ARSBA          ;SET UP CURRENT ADDRESS
3025          016146 012777 177777 162734          MOV          #-1, ARSWC          ;LOAD WC WITH -1
3026          016154 012777 000071 162722          MOV          #71, ARSCS1          ;DO READ FUNCTION
3027          016162 032777 000001 162714          BIT          #1, ARSCS1          ;DID GO BIT CLEAR
3028          016170 001401          BEQ          .+4          ;YES
3029          016172 104001          HLT          !CS1          ;GO BIT SHOULD BE CLEARED
3030          016174 105777 162704          TSTB          ARSCS1          ;WAIT FOR READY
3031          016200 100375          BPL          1$          ;WAIT
3032          016202 022777 144270 162674          CMP          #144270, ARSCS1          ;DID ERRS CLEAR BY SETTING GO BIT?
3033          016210 001401          BEQ          .+4          ;YES
3034          016212 104002          HLT          !ER          ;NO

```


MO6

MAINDEC-11-DZRSB-E RH11-RS03LA-RS03-RS04 BASIC FUNCTION DIAGNOSTIC MACY11 27(732) 25-SEP-76 09:06 PAGE 78
DZRSBE.P11 TST105 EXECUTE FUNCTION WITH ERROR BITS SET

3035	016214	017700	162666		MOV	DRSCS2,BAD	;GET CS2
3036	016220	012701	001100		MOV	#1100,GOOD	;GET CORRECT ANS
3037	016224	053701	001152		BIS	UNNUM,GOOD	;GET UNIT #
3038	016230	020100			CMP	GOOD,BAD	;IS CS2 CORRECT?
3039	016232	001401			BEQ	.+4	;YES
3040	016234	104000			HLT		;GOOD = WHAT CS2 SHOULD =
3041	016236	022777	150600	162652	CMP	#150600,DRSDS	;DID ERR BITS SET?
3042	016244	001401			BEQ	.+4	;NO
3043	016246	104040			HLT	!DS	;ERR BIT SHOULD BE 1
3044	016250	022777	177777	162632	CMP	#-1,DRSWC	;DID WC MOVE?
3045	016256	001401			BEQ	.+4	;NO
3046	016260	104010			HLT	!WC	;WC SHOULD = 1777777
3047	016262	005777	162626		TST	DRSDA	;DID DA MOV
3048	016266	001401			BEQ	.+4	;NO
3049	016270	104004			HLT	!DA	;DA SHOULD =0
3050	016272	023777	001102	162612	CMP	DRBUFSV,DRSBA	;DID BA MOVE
3051	016300	001401			BEQ	.+4	;NO
3052	016302	104020			HLT	!BA	;BA MOVED
3053	016304	033777	001156	162610	BIT	UNCMP,DRSAS	;AS SHOULD BE SET
3054	016312	001001			BNE	.+4	;IS IT?
3055	016314	104100			HLT	!AS	;NO
3056	016316	022777	177017	162574	CMP	#177017,DRSER	;DID ER CHANGE?
3057	016324	001401			BEQ	.+4	;NO
3058	016326	104002			HLT	!ER	;ER SHOULD NOT CHANGE


```

3059                                     ;*****
3060                                     ;TEST 106                                PAT AND MCPE TEST
3061                                     ;*****
3062 016330 104400                       TST106: SCOPE
3063
3064 016332 104414                       PATST: CLRDK
3065 016334 052777 000020 162544       BIS      #BIT4,DRSCS2      ;CLEAR ALL RS REG
3066 016342 005777 162562               TST      DRSMR          ;SET PAT
3067 016346 017700 162534               MOV      DRSCS2,BAD      ;READ DRIVE REG
3068 016352 012701 000120               MOV      #120,GOOD      ;GET CS2
3069 016356 053701 001152               BIS      UNNUM,GOOD      ;MDPE SHOULD
3070 016362 020100                       CMP      GOOD,BAD      ;NOT SET
3071 016364 001401                       BEQ      .+4            ;IS CS2 CORRECT?
3072 016366 104000                       HLT
3073 016370 012777 000010 162532       MOV      #10,DRSMR      ;YES
3074 016376 022777 000010 162514       CMP      #10,DRSER      ;BAD = CS2 GOOD = CORRECT ANS
3075 016404 001401                       BEQ      .+4            ;CAUSE PAR TO SET IN RSER
3076 016406 104140                       HLT      !AS!DS        ;DID PAR SET?
3077 016410 017700 162506               MOV      DRASAS,BAD      ;YES
3078 016414 013701 001156               MOV      UNCMP,GOOD      ;GET AS REG
3079 016420 042701 177400               BIC      #177400,GOOD    ;GET UNIT ATA BIT
3080 016424 020100                       CMP      GOOD,BAD      ;CLEAR JUNK
3081 016426 001401                       BEQ      .+4            ;IS AS REG CORRECT?
3082 016430 104100                       HLT      !AS          ;YES
3083 016432 022777 104200 162444       CMP      #104200,DRSCS1 ;AS BIT SHOULD BE SET
3084 016440 001401                       BEQ      .+4            ;DID ERRS SET IN CS1?
3085 016442 104040                       HLT      !DS          ;YES
3086 016444 104414                       CLRDK
3087 016446 022777 004200 162430       CMP      #4200,DRSCS1  ;CS1 SHOULD =104200
3088 016454 001401                       BEQ      .+4            ;CLEAR RS REG
3089 016456 104002                       HLT      !ER          ;IS CS1 CORRECT?
3090 016460 017700 162422               MOV      DRSCS2,BAD      ;CLEARING ATA
3091 016464 013701 001152               MOV      UNNUM,GOOD      ;NO
3092 016470 052701 000100               BIS      #100,GOOD      ;CHECK TO SEE
3093 016474 020100                       CMP      GOOD,BAD      ;IF PAT CLEARS
3094 016476 001401                       BEQ      .+4
3095 016500 104000                       HLT
;PAT DID NOT CLEAR

```



```

*****
;TEST 107                               SET PAT BIT AND LOAD FUNCTION
*****
TST107: SCOPE

SETPAT: CLDRK
        BIS      #BAI, @RSCS2
        MOV      #177777, OUTBUF
        MOV      @#0BUFSV, @RSBA
        MOV      #-1000, @RSMC
        BIS      #BIT4, @RSCS2
        MOV      #71, @RSCS1
15:     TSTB     @RSCS1
        BPL      15
        CMP      #144270, @RSCS1
        BEQ      .+4
        HLT      !DS
        CMP      @#0BUFSV, @RSBA
        BEQ      .+4
        HLT      !BA
        CMP      #-1000, @RSMC
        BEQ      .+4
        HLT      !WC
;CLEAR ALL REG
;SET BAI
;DATA TO BE XFERED
;SET UP CURRENT ADDRESS
;LOAD WC WITH -1
;SET PAT BIT
;DO READ FUNCTION
;WAIT FOR READY
;WAIT
;DID CS1 GET LOADED?
;NO
;IT SHOULD NOT
;DID BA MOVE?
;NO
;YES
;DID WC MOVE?
;NO
;YES WHY?

*****
;TEST 110                               DO FUNCTION THEN SET PAT BIT
*****
TST110: SCOPE
FUNDO:  CLDRK
        BIS      #BAI, @RSCS2
        MOV      #177777, OUTBUF
        MOV      @#0BUFSV, @RSBA
        MOV      #-1000, @RSMC
35:     MOV      #71, @RSCS1
        BMI      35
        BIS      #BIT4, @RSCS2
25:     TSTB     @RSCS1
        BPL      25
        CMP      #144270, @RSCS1
        BEQ      .+4
        HLT      !ER
        MOV      @RSCS2, BAD
        MOV      #730, GOOD
        BIS      UNNUM, GOOD
        CMP      GOOD, BAD
        BEQ      .+4
        HLT
;CLEAR ALL REG
;SET BAI
;DATA TO BE XFERED
;SET UP CURRENT ADDRESS
;LOAD WC WITH -1
;DO A READ
;WAIT FOR BUSY
;SET PAT
;WAIT FOR READY
;DID MCPE SET?
;NO
;YES
;GET CS2
;GET CORRECT ANS
;GET UNIT #
;IS CS2 CORRECT?
;YES
;GOOD = WHAT CS2 SHOULD =

```


3143	016732	022777	010600	162156	CMP	#10600, JRSDS	:DID ERR BITS SET?
3144	016740	001401			BEQ	.+4	:NO
3145	016742	104040			HLT	:DS	:ERR BIT SHOULD BE 1
3146	016744	005777	162150		TST	JRSE	:IS ER CLEAR?
3147	016750	001401			BEQ	.+4	:YES
3148	016752	104044			HLT	:DS!DA	:NO ERRORS SHOULD BE SET
3149	016754	104414			CLRDK		:CLEAR ALL RS REG
3150	016756	022777	004200	162120	CMP	#4200, JRSCS1	:IS CS1 CORRECT?
3151	016764	001401			BEQ	.+4	:YES
3152	016766	104040			HLT	:DS	
3153							:*****
3154							:TEST 111
3155							:TEST PAR BY SETTING PAT
3156	016770	104400					:*****
3157							:TST111: SCOPE
3158	016772	104414			PATTST: CLRDK		:CLEAR ALL RS REG
3159	016774	052777	000010	162104	BIS	#BA1, JRSCS2	:SET BAI
3160	017002	012737	177777	027364	MOV	#177777, OUTBUF	:DATA TO BE XFERED
3161	017010	013777	001102	162074	MOV	#0BUFSV, JRSCS1	:SET UP CURRENT ADDRESS
3162	017016	012777	177000	162064	MOV	#-1000, JRSCS1	:LOAD MC WITH -1
3163	017024	012777	000061	162052	MOV	#61, JRSCS1	:DO WRITE FUNCTION
3164	017032	105777	162046		TSTB	JRSCS1	:WAIT FOR READY
3165	017036	100775			BMI	15	:WAIT
3166	017040	052777	000020	162040	BIS	#BIT4, JRSCS2	:SET PAT
3167	017046	105777	162032		TSTB	JRSCS1	:WAIT FOR READY
3168	017052	100375			BPL	25	
3169	017054	022777	144260	162022	CMP	#144260, JRSCS1	:DID MCPE SET?
3170	017062	001401			BEQ	.+4	:NO
3171	017064	104002			HLT	:ER	:YES
3172	017066	017700	162014		MOV	JRSCS2, BAD	:GET CS2
3173	017072	012701	000230		MOV	#230, GOOD	:GET CORRECT ANS-- DO NOT CHECK IR - REASON 50 CYCLE
3174	017076	053701	001152		BIS	UNNUM, GOOD	:GET UNIT #
3175	017102	042700	000100		BIC	#BIT6, BAD	:CLEAR IR BIT FOR CS2 COMPARE
3176	017106	020100			CMP	GOOD, BAD	:IS CS2 CORRECT?
3177	017110	001401			BEQ	.+4	:YES
3178	017112	104000			HLT		:GOOD = WHAT CS2 SHOULD =
3179	017114	022777	150600	161774	CMP	#150600, JRSDS	:DID ERR BITS SET?
3180	017122	001401			BEQ	.+4	:NO
3181	017124	104040			HLT	:DS	:ERR BIT SHOULD BE 1
3182	017126	022777	000010	161764	CMP	#10, JRSE	:DID PAR SET?
3183	017134	001401			BEQ	.+4	:YES
3184	017136	104044			HLT	:DS!DA	:NO
3185	017140	104414			CLRDK		:CLEAR ALL RS REG
3186	017142	022777	004200	161734	CMP	#4200, JRSCS1	:IS CS1 CORRECT?
3187	017150	001401			BEQ	.+4	:YES
3188	017152	104040			HLT	:DS	
3189	017154	005777	161740		TST	JRSE	:DID PAR CLEAR?
3190	017160	001401			BEQ	.+4	:YES
3191	017162	104040			HLT	:DS	


```

3192                                     ;*****
3193                                     ;TEST 112                                TEST THE ABILITY TO FILL THE LAST SECTOR
3194                                     ;*****
3195 017164 104400                       TST112: SCOPE
3196
3197 017166 104414                       LASTSC: CLRDK
3198 017170 012777 007777 161716        MOV      #7777, @RSDA
3199 017176 022737 000004 001162        CMP      #4, @RSDA
3200 017204 001004                BNE      2$
3201 017206 012777 177740 161674        MOV      #-40, @RSWC
3202 017214 000411                BR       1$
3203 017216 012777 177700 161664 2$:    MOV      #-100, @RSWC
3204 017224 005737 001162                TST      @RSDA
3205 017230 001403                BEQ      1$
3206 017232 012777 177600 161650        MOV      #-200, @RSWC
3207 017240 013777 001102 161644 1$:    MOV      @#0BUF5V, @RSBA
3208 017246 012777 000061 161630        MOV      #61, @RSCS1
3209 017254 004737 026600                JSR      PC, WAITRY
3210 017260 104001                HLT      !CS1
3211 017262 005777 161632                TST      @RSE
3212 017266 001401                BEQ      +4
3213 017270 104002                HLT      !ER
3214 017272 022777 012600 161616        CMP      #12600, @RSDS
3215 017300 001401                BEQ      +4
3216 017302 104040                HLT      !DS
3217 017304 005777 161574                TST      @RSCS1
3218 017310 100001                BPL      +4
3219 017312 104001                HLT      !CS1
3220 017314 104414                CLRDK
3221 017316 022777 010600 161572        CMP      #10600, @RSDS
3222 017324 001401                BEQ      +4
3223 017326 104040                HLT      !DS

;CLEAR ALL RS REG
;SET RSDA=TO ALL ONES
;LA DISK?
;NO
;LOAD WORD COUNT
;CONTINUE
;WORD COUNT=-100
;IS THIS A RS04?
;NO
;YES
;CURRENT ADDRESS=OUTBUF
;WRITE
;WAIT FOR READY
;RDY DID NOT SET
;DID ANY ERROR BITS SET?
;NO
;GOT AN ERROR
;DID LBT SET?
;YES
;LBT DID NOT SET
;IS ERROR FLAG SET
;ERROR IS SET
;SC DID NOT SET
;CLEAR ALL RS REG
;DID ATA +LBT CLEAR
;YES
;ATA DID NOT CLEAR BY CLR BIT

```


3224
3225
3226
3227
3228
3229
3230
3231
3232
3233
3234
3235
3236
3237
3238
3239
3240
3241
3242
3243
3244
3245
3246
3247
3248
3249
3250
3251
3252
3253
3254
3255
3256
3257
3258
3259
3260
3261
3262
3263
3264
3265
3266
3267
3268
3269
3270
3271
3272
3273
3274
3275
3276
3277

017330 104400
017332 104414
017334 012737 177777 027364
017342 013777 001102 161542
017350 022737 000004 001162
017356 001004
017360 012777 177740 161522
017366 000411
017370 012777 177600 161512 4S:
017376 005737 001162
017402 001003
017404 012777 177700 161476
017412 052777 000010 161466 5S:
017420 012777 000061 161456
017426 105777 161452 3S:
017432 100375
017434 005077 161454
017440 012737 177777 027364
017446 013777 001102 161436
017454 012777 177777 161426
017462 052777 000010 161416
017470 012777 000061 161406
017476 105777 161402 1S:
017502 100375
017504 042777 000010 161374
017512 005737 001162
017516 001404
017520 012737 000200 001174
017526 000403
017530 012737 000100 001174 7S:
017536 013701 001102 8S:
017542 012721 177777
017546 005021 6S:
017550 005337 001174
017554 001374
017556 005077 161332
017562 013777 001102 161322
017570 022737 000004 001162
017576 001004
017600 012777 177740 161302
017606 000412
017610 005737 001162 11S:
017614 001404
017616 012777 177600 161264
017624 000403
017626 012777 177700 161254 9S:
017634 012777 000051 161242 10S:

;FILL SECTOR WITH ALL ONES.
;NOW WRITE 1ST WORD IN SECTOR
;TEST REMAINING 63 WORDS FOR 0

;TEST 113 TEST FOR ZERO'S IN PARTIAL FILLED SECTOR

TST113: SCOPE
SECT: CLRDK

MOV #1,OUTBUF
MOV #0,OUTFSV,RSBA
CMP #4,RS04DT
BNE 4S
MOV #40,RSWC
BR 5S
MOV #200,RSWC
TST RS04DT
BNE 5S
MOV #100,RSWC
BIS #BAI,RS0CS2
MOV #61,RS0CS1
TSTB RS0CS1
BPL 3S
CLR RS0DA
MOV #1,OUTBUF
MOV #0,OUTFSV,RSBA
MOV #1,RSWC
BIS #10,RS0CS2
MOV #61,RS0CS1
TSTB RS0CS1
BPL 1S
BIC #10,RS0CS2
TST RS04DT
BEQ 7S
MOV #200,WORK
BR 8S
MOV #100,WORK
MOV #0,OUTFSV,R1
MOV #1,(R1)+
CLR (R1)+
DEC WORK
BNE 6S
CLR RS0DA
MOV #0,OUTFSV,RSBA
CMP #4,RS04DT
BNE 11S
MOV #40,RSWC
BR 10S
TST RS04DT
BEQ 9S
MOV #200,RSWC
BR 10S
MOV #100,RSWC
MOV #51,RS0CS1

;CLEAR ALL RS REG
;PUT -1 INTO OUTBUF
;SET UP CURRENT ADDR
;RS03LA DISK?
;NO
;LOAD WORD COUNT
;CONTINUE
;LOAD WC FOR RS04
;RS04?
;YES
;SET WORD COUNT TO -100
;SET BAI BIT
;WRITE
;IS RDY SET?
;NO
;SET DSK ADDRESS TO 0
;PUT 177777 INTO OUTBUF
;SET UP CURRENT ADDR
;SET WORD COUNT TO -1
;SET BAI BIT
;WRITE
;IS RDY SET?
;NO
;CLEAR BAI BIT
;RS04?
;NO
;YES
;CONT
;SET UP BUFFER
;GET STARTING ADD OF BUF
;LOAD FIRST WD WITH -1
;LOAD REST WITH 0
;DONE YET?
;NO
;SET DSK ADDRESS TO 0
;SET UP CURRENT ADDR
;RS03LA DISK?
;NO
;LOAD WORD COUNT
;CONTINUE
;RS04?
;NO
;YES
;CONT
;SET WORD COUNT TO -100
;WRITE CHECK

F07

MAINDEC-11-DZRSB-E
DZRSBE.P11

TST113

RH11-R503LA-R503-R504 BASIC FUNCTION DIAGNOSTIC MACY11 27(732) 25-SEP-76 09:06 PAGE 84
TEST FOR ZERO'S IN PARTIAL FILLED SECTOR

```

3278 017642 032777 000200 161234 2$: BIT #200, JRSCS1 ;IS RDY SET?
3279 017650 001774 BEQ 2$ ;NO
3280 017652 013701 001152 MOV UNNUM, GOOD ;GET UNIT #
3281 017656 052701 000100 BIS #100, GOOD ;SET IR BIT
3282 017662 017700 161220 MOV JRSCS2, BAD ;GET CS2
3283 017666 020100 CMP GOOD, BAD ;IS CS2 CORRECT?
3284 017670 001401 BEQ +4 ;YES
3285 017672 104002 HLT !ER ;THERE WAS A WRITE CHECK ERROR

```

```

*****
;TEST 114 IF MEMORY MANAGEMENT IS AVAILABLE CHECK THE EXTENDED MEMORY ADDR
*****
TST114: SCOPE

```

```

3292 017676 104414 EXTTST: CLDK ;CLEAR ALL RS REG.
3293 017700 013737 022414 001164 MOV TIMES, TIMSV ;SAVE LOOP #
3294 017706 012737 000010 022414 MOV #10, TIMES ;LOOP 10 TIMES
3295 017714 012737 020420 000004 MOV #EXTTRP, 4 ;SETUP TIMEOUT TRAP
3296 017722 012737 000340 000006 MOV #340, 6
3297 017730 005737 177572 TST @SR0 ;IF MEMORY MANAGEMENT IS NOT
3298 ;AVAILABLE THE PROGRAM WILL TRAP
3299 ;AND TRANSFER TO END OF THE TEST
3300 017734 012737 020412 000004 MOV #EXTTRP, 4
3301 017742 012737 007600 172356 MOV #7600, @KIPAR7 ;OPEN I/O REGISTERS
3302 017750 005037 172340 CLR @KIPAR0 ;FREE FIRST 4K
3303 017754 012737 000200 172342 MOV #200, @KIPAR1 ;ENABLE SECOND 4K
3304 017762 012737 002000 172344 MOV #2000, @KIPAR2
3305 017770 012737 177406 172300 MOV #400*256.-400+UP+RW, @KIPDR0 ;SET KIPDR0=RW UP 400 BLOCKS
3306 017776 012737 177406 172302 MOV #400*256.-400+UP+RW, @KIPDR1 ;SET KIPDR1=RW UP 400 BLOCKS
3307 020004 012737 177406 172304 MOV #400*256.-400+UP+RW, @KIPDR2 ;SET KIPDR2=RW UP 400 BLOCKS
3308 020012 012737 177406 172316 MOV #400*256.-400+UP+RW, @KIPDR7 ;SET KIPDR7=RW UP 400 BLOCKS
3309 020020 012737 000001 177572 MOV #1, @SR0 ;TURN ON MEMORY MANAGEMENT
3310 020026 012702 040000 MOV #40000, R2 ;R2 EQUALS BASE ADDR

```


G07

MAINDEC-11-DZRSB-E
DZRSBE.P11

TST114

RH11-RS03LA-RS03-RS04 BASIC FUNCTION DIAGNOSTIC MACY11 27(732) 25-SEP-76 09:06 PAGE 85
IF MEMORY MANAGEMENT IS AVAILABLE CHECK THE EXTENDED MEMORY ADDRESS BITS

3311	020032	012712	177777		7\$:	MOV	#177777, (R2)	: INSERT PATTERN INTO 200000
3312	020036	012777	177776	161044		MOV	#-2, ARSWC	: SETUP WORDCOUNT
3313	020044	012777	177777	161040		MOV	#177777, ARSBA	: SETUP BUS ADDR
3314	020052	012777	000061	161024		MOV	#61, ARSCS1	: WRITE TWO WORDS ON DISK. RSBA
3315								: STARTS AT 177777 TO FORCE CARRY
3316								: TO SET A16
3317	020060	105777	161020			TSTB	ARSCS1	: WAIT FOR READY
3318	020064	100375				BPL	-4	
3319	020066	005777	161012			TST	ARSCS1	
3320	020072	100002				BPL	1\$	
3321	020074	104046				HLT	!ER!DA!DS	: STATUS ERROR AFTER 2 WORD WRITE
3322	020076	000447				BR	2\$	: USING MEXD
3323	020100	022777	004660	160776	1\$:	CMP	#4660, ARSCS1	: IS CS1 CORRECT
3324	020106	001402				BEQ	3\$	: YES
3325	020110	104002				HLT	!ER	: CS2 DID NOT COMPARE
3326	020112	000441				BR	2\$	
3327	020114	005012			3\$:	CLR	(R2)	: CLEAR LOCATION 200000
3328	020116	005077	160772			CLR	ARSDA	: SETUP DA
3329	020122	012777	177777	160762		MOV	#177777, ARSBA	: SETUP BA
3330	020130	012777	177776	160752		MOV	#-2, ARSWC	: SETUP WC
3331	020136	012777	000071	160740		MOV	#71, ARSCS1	: READ TWO WORDS INTO LOCATIONS
3332								: 177777 AND 200000.
3333	020144	105777	160734			TSTB	ARSCS1	: WAIT FOR READY
3334	020150	100375				BPL	-4	
3335	020152	005777	160726			TST	ARSCS1	: ANY ERRORS?
3336	020156	100002				BPL	4\$	: BRANCH IF NO
3337	020160	104002				HLT	!ER	: ERROR AFTER READING 2 WORDS
3338	020162	000415				BR	2\$	
3339	020164	022777	004670	160712	4\$:	CMP	#4670, ARSCS1	: IS CS1 CORRECT?
3340	020172	001402				BEQ	5\$	: YES
3341	020174	104002				HLT	!ER	: CS1 DID NOT COMPARE
3342	020176	000407				BR	2\$	: READ STARTING AT 177777
3343	020200	022712	177777		5\$:	CMP	#177777, (R2)	: WAS DATA READ INTO LOCATION
3344	020204	001404				BEQ	2\$	: 200000 CORRECTLY? - BRANCH IF YES
3345	020206	012701	177777			MOV	#177777, GOOD	
3346	020212	011200				MOV	(R2), BAD	
3347	020214	104000				HLT		: DATA COMPARE ERROR AT 200000
3348	020216	000240			2\$:	NOP		

MAINDEC-11-DZRSB-E RH11-RS03LA-RS03-RS04 BASIC FUNCTION DIAGNOSTIC MACY11 27(732) 25-SEP-76 09:06 PAGE 86
DZRSBE.P11 TST114 IF MEMORY MANAGEMENT IS AVAILABLE CHECK THE EXTENDED MEMORY ADDRESS BITS

MAINDEC-11-DZRSB-E RH11-RS03LA-RS03-RS04 BASIC FUNCTION DIAGNOSTIC MACY11 27(732) 25-SEP-76 09:06 PAGE 86
DZRSBE.P11 TST114 IF MEMORY MANAGEMENT IS AVAILABLE CHECK THE EXTENDED MEMORY ADDRESS BITS

3349	020220	104414			EXTT1:	CLDRK		;CLEAR ALL REG
3350	020222	012737	004000	172344		MOV	#4000,R2	
3351	020230	012702	040000			MOV	#40000,R2	;R2 EQUALS THE BASE ADDR
3352	020234	012712	177777		7\$:	MOV	#177777,(R2)	;INSERT PATTERN INTO 400000
3353	020240	012777	177777	160644		MOV	#177777,ARSA	;SETUP BUS ADDR
3354	020246	012777	177776	160634		MOV	#-2,ARSWC	;LOAD WC
3355	020254	012777	000461	160622		MOV	#461,ARSCSI	;SET BIT A16 AND WRITE
3356	020262	105777	160616			TSTB	ARSCSI	;WAIT FOR READY
3357	020266	100375				BPL	-4	
3358	020270	005777	160610			TST	ARSCSI	;ANY ERRORS?
3359	020274	100001				BPL	4\$	;BRANCH IF NO
3360	020276	104002				HLT	!ER	;ERROR AFTER READING 2 WORDS
3361	020300	022777	005260	160576	4\$:	CMP	#5260,ARSCSI	;IS CS1 CORRECT?
3362	020306	001401				BEQ	10\$	;BRANCH IF YES
3363	020310	104002				HLT	!ER	;NO CS1 DID NOT COMPARE
3364								;READ STARTING AT 377777
3365	020312	005012			10\$:	CLR	(R2)	;CLEAR LOCATION 400000
3366								;READ TWO WORDS STARTING AT 377777
3367	020314	012777	177776	160566		MOV	#-2,ARSWC	;SETUP WC
3368	020322	005077	160566			CLR	ARSDA	;SETUP DA
3369	020326	012777	177777	160556		MOV	#177777,ARSA	
3370	020334	012777	000471	160542		MOV	#471,ARSCSI	;CLEAR A 17 SET A16, READ
3371	020342	105777	160536			TSTB	ARSCSI	;WAIT FOR READY
3372	020346	100375				BPL	-4	
3373	020350	005777	160530			TST	ARSCSI	;ANY ERRORS?
3374	020354	100002				BPL	11\$	;BRANCH IF NO
3375	020356	104002				HLT	!ER	;ERROR WHILE READING TWO WORDS
3376	020360	000414				BR	EXTRP	
3377	020362	022777	005270	160514	11\$:	CMP	#5270,ARSCSI	;IS CS1 CORRECT?
3378	020370	001401				BEQ	12\$	;BRANCH IF YES
3379	020372	104002				HLT	!ER	;CS1 DID NOT COMPARE
3380								;READ STARTING AT 377777
3381	020374	022712	177777		12\$:	CMP	#177777,(R2)	;WAS DATA READ INTO LOCATION 400000
3382	020400	001404				BEQ	EXTRP	;CORRECTLY? - BRANCH IF YES
3383	020402	012701	177777			MOV	#177777,GOOD	
3384	020406	011200				MOV	(R2),BAD	
3385	020410	104000				HLT		
3386								;DATA COMPARE ERROR AT 400000 IF
3387	020412	005037	177572		EXTRP:	CLR	ASRO	;RECEIVED=0 - LOCATION WASN'T ACCESSED
3388	020416	000401				BR	EXT1	;TURN OFF MEMORY MANAGEMENT
3389	020420	000240			EXTTRP:	NOP		;UPDATE TEST NUMBERS
3390	020422	012706	000500		EXT1:	MOV	#500,SP	;RESTORE STACK
3391	020426	012737	000006	000004	MEMOUT:	MOV	#6,4	
3392	020434	005037	000006			CLR	6	


```

3393                                     ;*****
3394                                     ;TEST 115                                TEST PROGRAM INTERRUPT BY MOVING 300 INTO RSCS1
3395                                     ;*****
3396 020440 104400                       TST115: SCOPE
3397
3398 020442 104414                       QES:  CLRDK                                ;CLEAR ALL DRIVES
3399 020444 012706 000500                MOV    #500,SP                        ;SETUP STACK
3400 020450 012777 020522 160456        MOV    #PGTRAP,RSVEC                ;SET UP VECTOR
3401 020456 012777 000340 160452        MOV    #340,RSVCPS                 ;SET TRAP PS
3402 020464 012737 000200 177776        MOV    #200,PS                     ;SET PS AT PRIORITY 4
3403 020472 012777 000300 160404        MOV    #300,RSRCS1                 ;THIS SHOULD CAUSE A TRAP
3404 020500 012737 000500 001174        MOV    #500,WORK                     ;SETUP LOOP
3405 020506 005337 001174                IS:  DEC    WORK                     ;DEC LOOP SHOULD
3406 020512 001375                        BNE    IS                          ;INTERRUPT BEFORE LOOP IS DONE
3407 020514 104001                        HLT    !CS1                       ;SHOULD NEVER GET HERE
3408 020516 000137 020536                JMP    QESDON                      ;GET OUT
3409
3410 020522 022626 004200 160352        PGTRAP: CMP    (6)+(6)+                ;TRAP OK
3411 020524 022777 004200 160352        CMP    #4200,RSRCS1                ;DID IE CLEAR?
3412 020532 001401                        BEQ    +4                          ;YES
3413 020534 104001                        HLT    !CS1                       ;IE SHOULD BE CLEARED
3414 020536
3415
3416                                     ;*****
3417                                     ;TEST 116                                TEST THAT DISK DOES NOT INTERRUPT WHEN PS IS AT 5
3418                                     ;*****
3419 020536 104400                       TST116: SCOPE
3420
3421 020540 012706 000500                INTR5: MOV    #500,SP                ;SETUP STACK
3422 020544 013737 001164 022414        MOV    TMSV,TIMES                ;RESTORE LOOP COUNTER
3423 020552 104414                        CLRDK                                ;CLEAR ALL RS REG
3424 020554 012777 020634 160352        MOV    #INT112,RSVEC                ;SET UP INTERRUPT VECTOR
3425 020562 012777 000340 160346        MOV    #340,RSVCPS                 ;SET PRIO.
3426 020570 012737 000240 177776        MOV    #240,PS                     ;LOCK OUT ALL INTERRUPTS ABOVE
3427 020576 013700 177776                MOV    PS,BAD                     ;GET PS
3428 020602 012777 177777 160300        MOV    #177777,RSWC                ;SET WORD COUNT TO -1
3429 020610 013777 001102 160274        MOV    #0BUFSV,RSBA                ;LOAD CURRENT ADDRESS
3430 020616 012777 000161 160260        MOV    #161,RSRCS1                ;GO WRITE (INTERRUPT ENABLED)
3431 020624 004737 026600                JSR    PC,WAITRY                ;WAIT FOR READY
3432 020630 104001                        HLT    !CS1                       ;NO RDY NEVER CAME UP
3433 020632 000403                        BR     INTDON                      ;RESTART ROUTINE
3434
3435                                     ;PROCESSOR SHOULD NOT TRAP TO INT112
3436 020634 012701 000240                INT112: MOV    #240,GOOD                ;WHAT PS SHOULD HAVE
3437 020640 104000                        HLT                                ;GOOD = CORRECT ANS FOR PS
3438 020642                                     INTDON:                                     ;DONE GET OUT

```



```

3439 ;*****
3440 ;TEST 117 TEST THAT DISK DOES INTERRUPT WHEN PS IS AT 4
3441 ;*****
3442 020642 104400 TST117: SCOPE
3443
3444 020644 012706 000500 INTR4: MOV #500,SP ;SETUP STACK
3445 020650 104414 CLRDK ;CLEAR ALL RS REG
3446 020652 012777 020746 160254 MOV #INT114,RSVEC ;SET UP DISK TRAP VECTOR
3447 020660 012777 000340 160250 MOV #340,RSVCPS ;SET PRIO.
3448 020666 012737 000200 177776 MOV #200,PS ;SET PROCESSOR TO PRIORITY 4
3449 020674 013700 177776 MOV PS,BAD ;GET PS
3450 020700 012701 000200 MOV #200,GOOD ;GET CORRECT PS
3451 020704 012777 177777 160176 MOV #177777,RSWC ;SET WORD COUNT TO -1
3452 020712 013777 001102 160172 MOV #0BUFSV,RSBA ;LOAD CURRENT ADDRESS
3453 020720 012777 000161 160156 MOV #161,RSCSI ;WRITE (INTERRUPT ENABLE)
3454 020726 005037 001174 CLR WORK
3455 020732 005237 001174 INC WORK ;WAIT FOR INTERRUPT TO OCCUR
3456 020736 001375 BNE .-4
3457 020740 104000 HLT ;GOOD=CORRECT PS BAD=WRONG PS
3458 020742 104042 HLT !ER!DS
3459 020744 000405 BR DONINT ;CONT
3460 020746 022777 004260 160130 INT114: CMP #4260,RSCSI ;DID IE CLEAR?
3461 020754 001401 BEQ +4 ;YES
3462 020756 104001 HLT !CSI ;WHY DID NOT IE CLEAR
3463 020760 DONINT:
3464
3465 ;*****
3466 ;TEST 120 TEST INTERRUPT ON ERROR
3467 ;*****
3468 020760 104400 TST120: SCOPE
3469
3470 020762 012706 000500 ERINT: MOV #500,SP ;SETUP STACK
3471 020766 012737 000200 177776 MOV #200,PS ;SET PS AT PRI 4
3472 020774 012777 021052 160132 MOV #ERRINT,RSVEC ;SET UP INTERRUPT ADD.
3473 021002 104414 CLRDK ;CLEAR ALL RS REG
3474 021004 012777 000340 160124 MOV #340,RSVCPS ;SET PRIO.
3475 021012 012777 177777 160074 MOV #177777,PSDA ;SET PSDA=TO ALL ONES
3476 021020 012777 177600 160062 MOV #177600,RSWC ;WORD COUNT=-200
3477 021026 013777 001102 160056 MOV #0BUFSV,RSBA ;CURRENT ADDRESS=OUTBUF
3478 021034 012777 000161 160042 MOV #161,RSCSI ;WRITE
3479 021042 004737 026600 JSR PC,WAITRY ;WAIT FOR READY
3480 021046 104042 1S: HLT !ER!DS ;Y DIDN'T PGM INTERRUPT IS RDY SET?
3481 021050 000406 BR FINTST ;GET OUT
3482 021052 022777 144260 160024 ERRINT: CMP #144260,RSCSI ;IS CSI RIGHT?
3483 021060 001401 BEQ +4 ;YES
3484 021062 104042 HLT !ER!DS
3485 021064 022626 CMP (6)+,(6)+ ;CLEAR STACK
3486 021066 FINTST:

```


K07

MAINDEC-11-DZRSB-E
DZRSBE.P11

TST121

RH11-RS03LA-RS03-RS04
DYNAMIC FUNCTION TEST

BASIC FUNCTION DIAGNOSTIC MACY11 27(732) 25-SEP-76 09:06 PAGE 89

```

3487      ;*****
3488      ;TEST 121          DYNAMIC FUNCTION TEST
3489      ;*****
3490      021066  104400
3491      TST121: SCOPE
3492      ;EXECUTE FUNCTION MODIFY UNIT # AND DO A DRIVE SEARCH
3493      ;DRIVE SEARCH WILL ONLY BE DONE IF THERE ARE AT LEAST 2 DRIVES
3494      ;2ND DRIVE MAY NOT BE TESTED YET SO IF THIS TEST FAILS CHECK 2ND DRIVE
3495      ;BEFORE TRYING TO DEBUG THIS TEST
3496      021070  104414      MODNUM: CLRDK      ;CLEAR ALL RS REG
3497      021072  013737  022414  001164      MOV      TIMES,TIMSV      ;SAVE LOOP COUNT
3498      021100  012737  000010  022414      MOV      #10,TIMES      ;LOOP ONLY 10 TIMES
3499      021106  005037  001176      CLR      WORK1      ;CLEAR WORK LOC.
3500      021112  005003      CLR      R3
3501      021114  005004      CLR      R4
3502      021116  012702  022044      MOV      #DVTAB,R2      ;SETUP TABLE
3503      021122  012737  000401  001174      MOV      #401,WORK      ;SETUP TO TEST FOR MORE DRIVES
3504      021130  033737  001174  001154  7$:      BIT      WORK,UNITSV      ;IS DRIVE ON SYSTEM?
3505      021136  001403      BEQ      6$      ;NO
3506      021140  020437  001152      CMP      R4,UNNUM      ;IS THIS THE SAME DRIVE?
3507      021144  001017      BNE      8$      ;NO
3508      021146  005204      6$:      INC      R4      ;UPDATE 2ND UNIT #
3509      021150  000241      CLC
3510      021152  006137  001174      ROL      WORK      ;CHECK FOR NEXT DRIVE
3511      021156  103364      BCC      7$      ;NOT DONE YET
3512      021160  032737  000010  001176      BIT      #BIT3,WORK1      ;MULTI DRIVE?
3513      021166  001016      BNE      12$      ;YES
3514      021170  013705  001152      MOV      UNNUM,R5      ;LOAD UNIT NO
3515      021174  005205      INC      R5      ;CHANGE IT
3516      021176  042705  177770      BIC      #177770,R5      ;CLEAR JUNK
3517      021202  000410      BR      12$
3518      021204  052737  000010  001176  8$:      BIS      #BIT3,WORK1      ;SET FOUND MULTI DRIVE
3519      021212  010422      MOV      R4,(R2)+      ;LOAD UNIT # INTO TABLE
3520      021214  005203      INC      R3      ;COUNT # OF DRIVES
3521      021216  010337  001172      MOV      R3,SAVEE      ;SAVE IT
3522      021222  000751      BR      6$
3523      021224  012777  021340  157702  12$:      MOV      #TSTVEC,RSVEC      ;SETUP INT. TRAP
3524      021232  012777  000340  157676      MOV      #340,RSVCPS      ;SETUP PRIO.
3525      021240  005037  027364      CLR      OUTBUF      ;CLR TO READ INTO
3526      021244  013777  001102  157640      MOV      #0BUFV,RSBA      ;SET UP CURRENT ADDRESS
3527      021252  012777  177000  157630      MOV      #-1000,RSWC      ;SET WORD COUNT
3528      021260  012777  000060  157626  1$:      MOV      #60,RSDA      ;LOAD DA
3529      021266  012702  022044      MOV      #DVTAB,R2      ;GET TABLE
3530      021272  012777  000161  157604      MOV      #161,RS1      ;GO WRITE
3531      021300  005703      TST      R3      ;MORE THEN 1 DRIVE?
3532      021302  001003      BNE      13$      ;YES
3533      021304  010577  157576      MOV      R5,RS2      ;NO MODIFY UNIT #
3534      021310  000407      BR      14$

```


L07

MAINDEC-11-DZRSB-E
DZRSBE.P11

TST121

RH11-R503LA-R503-R504
DYNAMIC FUNCTION TEST

BASIC FUNCTION DIAGNOSTIC MACY11 27(732) 25-SEP-76 09:06 PAGE 90

3535	021312	012277	157570		13\$:	MOV	(R2)+, ARSCS2	:LOAD UNIT#
3536	021316	012777	000131	157560		MOV	#131, ARSCS1	:DO SEARCH
3537	021324	005303				DEC	R3	:DONE ALL DRIVES YET?
3538	021326	001371				BNE	13\$	:NO
3539	021330	012737	000200	177776	14\$:	MOV	#200, ARPS	:ENABLE INTERRUPTS
3540	021336	000001			WTDV:	WAIT		
3541	021340	013777	001152	157540	TSTVEC:	MOV	UNNUM, ARSCS2	:GET 1ST DRIVE
3542	021346	017700	157532			MOV	ARSCS1, BAD	:GET CS1
3543	021352	042700	100000			BIC	#BIT15, BAD	:CLEAR SC
3544	021356	012701	004260			MOV	#4260, GOOD	:GET CORRECT ANS
3545	021362	020100				CMP	GOOD, BAD	:IS CS1 CORRECT?
3546	021364	001402				BEQ	4\$	:NO! X-FER OK
3547	021366	104000				HLT		:CS1 SHOULD = 14270 OR 4270
3548	021370	104140				HLT	:DS!AS	
3549	021372	005777	157512		4\$:	TST	ARSWC	:TEST WC
3550	021376	001401				BEQ	.+4	:WORD COUNT DID OVERFLOW
3551	021400	104010				HLT	:WC	:SHOULD = 0
3552	021402	013701	001152			MOV	UNNUM, GOOD	:GET CORRECT
3553	021406	052701	000100			BIS	#100, GOOD	:ANS OF CS2
3554	021412	017700	157470			MOV	ARSCS2, BAD	:GET CS2
3555	021416	020100				CMP	GOOD, BAD	:IS CS2 CORRECT?
3556	021420	001401				BEQ	.+4	:YES
3557	021422	104000				HLT		:GOOD = CORRECT ANS FOR CS2
3558	021424	017700	157462			MOV	ARSB, BAD	:FETCH CURRENT ADDRESS
3559	021430	013701	001102			MOV	ARBUFSV, GOOD	:WHAT RSBA SHOULD EQUAL
3560	021434	062701	002000			ADD	#2000, GOOD	:UPDATE IT
3561	021440	020001				CMP	BAD, GOOD	:IS RSBA CORRECT
3562	021442	001401				BEQ	.+4	:YES EXECUTE CONTINUE
3563	021444	104000				HLT		:RSBA FAILED TO INCREMENT
3564	021446	022737	000004	001162		CMP	#4, RS04DT	:RS03LA?
3565	021454	001006				BNE	2\$	:NO
3566	021456	022777	000100	157430		CMP	#100, ARSDA	:IS DA CORRECT?
3567	021464	001420				BEQ	3\$	:OK
3568	021466	104104				HLT	:DA!AS	
3569	021470	000416				BR	3\$	:CONTINUE
3570	021472	005737	001162		2\$:	TST	RS04DT	:IS THIS A RS04?
3571	021476	001006				BNE	5\$	:YES
3572	021500	022777	000070	157406		CMP	#70, ARSDA	:IS DA CORRECT?
3573	021506	001407				BEQ	3\$	:YES
3574	021510	104104				HLT	:DA!AS	:DA NOT CORRECT
3575	021512	000405				BR	3\$	:CONTINUE
3576	021514	022777	000064	157372	5\$:	CMP	#64, ARSDA	:WAS RSDA INCREMENTED
3577	021522	001401				BEQ	.+4	:RSDA OK
3578	021524	104046				HLT	:DA!ER!DS	:RSDA SHOULD CONTAIN A 64
3579	021526	012777	040000	157350	3\$:	MOV	ARE, ARSCS1	:CLEAR ALL ERRORS IF ANY
3580	021534	032737	000010	001176		BIT	#BIT3, WORK1	:MULTI DRIVE?
3581	021542	001461				BEQ	WTDV1	:NO
3582	021544	012777	021570	157362	1\$:	MOV	ATTVEC, ARSVEC	:SETUP INT FOR NEXT DRIVE
3583	021552	012716	021566			MOV	WTDV2, (SP)	:GET WAIT
3584	021556	012777	000100	157320		MOV	#100, ARSCS1	:SET IE
3585	021564	000002				RTI		:RETURN
3586	021566	000001			WTDV2:	WAIT		


```

3587      ;SERVICE ROUTINE FOR SEARCH FUNCTIONS
3588      TTVEC: CLR R2 ;CLEAR UNIT #
3589      CLC
3590      MOV #401,WORK
3591      BIT WORK,ARSAS ;DID THIS DRIVE INT?
3592      BNE 2$ ;YES
3593      INC R2 ;UPDATE UNIT #
3594      CLC
3595      ROL WORK
3596      BCC 1$
3597      HLT !AS ;WHY DID WE INT WITH NO ATA???
3598      MOV R2,ARSCS2 ;GET DRIVE
3599      CMP #110600,ARSDS ;DID PIP CLEAR?
3600      BEQ +4 ;YES
3601      HLT !DS!AS ;PIP BIT DID NOT CLEAR
3602      CMP #104230,ARSCS1 ;DID SC SET?
3603      BEQ +4 ;YES
3604      HLT !AS!DS ;SC DID NOT SET
3605      DEC SAVEE ;COUNT # OF INT
3606      BEQ WTDV1 ;DONE YET?
3607      MOV WORK,ARSAS ;CLEAR AS
3608      MOV #100,ARSCS1 ;SET IE
3609      MOV #WTDV2,(SP) ;RETURN TO WAIT
3610      RTI
3611      MOV #340,#PS
3612      MOV #500,SP ;CLEAR STACK
3613      MOV RSVCPS,ARSVEC ;RESTORE INT VECTOR
3614      CLR ARSVCPS
3615      MODDON: SCOPE ;DONE
3616      MOV TIMSV,TIMES ;RESTORE LOOP COUNT
3617      MOV #340,#PS ;RESTORE PS
3618      MOV #1,ICNT ;FUGE TEST NUMBERS
3619      JMP #TRYNX ;TEST NEXT DRIVE
3620      ;SBTTL $DONE - BELL AND SCOPE ROUTINE
3621
3622      DONE: SCOPE ;TERMINATIONG SCOPE FOR LOOPING
3623      ADD #1,PCNT+2 ;ADD 1 TO THE PASS COUNT
3624      ADC PCNT ;MAKE IT DOUBLE PREC.
3625      BIT #SW10,ASWR ;RING THE BELL?
3626      BNE 4$ ;NO!
3627      TYPE +2 ;.ASCIZ <BELL><177>
3628      MOV #42,R0 ;GET MONITOR ADDRESS
3629      BEQ SEND1 ;IF NONE
3630      RESET
3631      SENDAD: JSR 7,(0) ;GO TO MONITOR
3632      240,240,240 ;SAVE ROOM FOR ACT11
3633      SEND1: JMP #BEGIN ;RETURN
3634
3635      .TBIT: 0 ;T BIT FLAG
3636
3637      DVTAB: .BLKW 10

```



```

3638      .SBTTL      $TYPE - TTY TYPEOUT ROUTINE
3639
3640      ; THIS ROUTINE IS USE TO TYPE ASCII MESSAGES ON THE TTY. THE
3641      ; CALL CAN BE IN ONE OF 3 FORMS: 1) "TYPE ADR" - TYPES THE
3642      ; MESSAGE STARTING IN LOCATION "ADR:" 2) "TYPE CHAR" - TYPES
3643      ; THE ASCII "CHAR", AND 3) "PRINT <<15><12>"MESSAGE">" - TYPES
3644      ; THE MESSAGE WHICH IS INLINE ASCII. THE FILLER CHARACTER WHICH IS
3645      ; TYPED AFTER A LINE FEED IS IN FILCHR AND THE NUMBER OF FILLERS
3646      ; IS IN FILCHR+1.
3647
3648      022064 010446      .TYPE:  MOV R4,-(6)      ;SAVE R4
3649      022066 010546      MOV R5,-(6)      ;SAVE R5
3650      022070 017605 000004      MOV @4(6),R5      ;GET ADDRESS TO BE TYPED
3651      022074 032705 177400      BIT #177400,R5      ;IS IT A TYPEN?
3652      022100 001002      BNE 1$      ;NO
3653      022102 016605 000004      MOV 4(6),R5      ;GET ADDRESS OF CHARACTER
3654      022106 105715      1$:  TSTB (R5)      ;TERMINATOR?
3655      022110 001423      BEQ 2$      ;GET OUT IF SO
3656      022112 122715 000012      CMPB #12,(R5)      ;IS THE CHAR A LINE FEED
3657      022116 001012      BNE 4$      ;NO - GET OUT
3658      022120 113704 001015      MOVB FILCHR+1,R4      ;GET THE FILL COUNT
3659      022124 113777 001014 156672 5$:  MOVB FILCHR,@TPB      ;TYPE A FILLER
3660      022132 105777 156660      TSTB @TPS      ;DONE YET?
3661      022136 100375      BPL -4      ;NO - WAIT
3662      022140 005304      DEC R4      ;DEC COUNT
3663      022142 001370      BNE 5$      ;LOOP UNTIL 0
3664      022144 112577 156654      4$:  MOVB (R5)+,@TPB      ;LOAD AND TYPE THE CHARACTER
3665      022150 105777 156642      TSTB @TPS      ;IS THE PRINTER READY
3666      022154 100375      BPL -4      ;WAIT UNTIL IT IS
3667      022156 000753      BR 1$      ;GET THE NEXT CHARACTER
3668      022160 017646 000004      2$:  MOV @4(6),-(6)      ;GET ADDRESS TO BE TYPED
3669      022164 062766 000002 000006      ADD #2,6(6)      ;ADD 2 TO THE ADDRESS
3670      022172 022666 000004      CMP (6)+,4(6)      ;IS IT .+2?
3671      022176 001006      BNE 3$      ;NO
3672      022200 062705 000002      ADD #2,R5      ;ADD 2 TO THE ADDRESS
3673      022204 042705 000001      BIC #1,R5      ;BACK UP TO AN EVEN BYTE
3674      022210 010566 000004      MOV R5,4(6)      ;RESTORE ADDRESS
3675      022214 012605      3$:  MOV (6)+,R5      ;RESTORE R5
3676      022216 012604      MOV (6)+,R4      ;RESTORE R4
3677      022220 000002      RTI      ;RETURN

```


Address	Instruction	Comment
3678	.SBTTL	
3679	SSCOPE - SCOPE LOOP HANDLER	
3680	;THIS ROUTINE HANDLES THE ITERATIONS, LOOPING, ERROR	
3681	;LOOPING, AND THE DISPLAYING OF THE TEST NUMBER.	
3682	;"SCOPE" IS PLACED BETWEEN EACH SUBTEST IN THE TEST AND	
3683	;RECORDS THE STARTING ADDRESS OF THE SUBTEST IN "LAD:"	
3684		
3685	022222 104416	.SCOPE: KBDIN
3686	022224 032777 000400 156574	BIT #SWB,JSWR
3687	022232 001404	1\$
3688	022234 127737 156566 001000	BEQ JSWR,ICNT
3689	022242 001453	.OVER
3690	022244 032777 040000 156554 1\$:	BIT #SW14,JSWR
3691	022252 001045	BNE .KIT
3692	022254 000416	BR 3\$
3693	022256 013746 000004	MOV J#4,-(6)
3694	022262 012737 022302 000004	MOV #4\$J#4
3695	022270 005737 177060	TST J#177060
3696	022274 012637 000004	MOV (6)+,J#4
3697	022300 000422	BR .SVLAD
3698	022302 022626 4\$:	CMP (6)+,(6)+
3699	022304 012637 000004	MOV (6)+,J#4
3700	022310 000426	BR .KIT
3701	022312 032777 004000 156506 3\$:	BIT #SW11,JSWR
3702	022320 001012	BNE .SVLAD
3703	022322 105737 001001	TSTB ICNT+1
3704	022326 001404	BEQ 2\$
3705	022330 123737 022414 001001	CMPB TIMES,ICNT+1
3706	022336 003013	BGT .KIT
3707	022340 112737 000001 001001 2\$:	MOVB #1,ICNT+1
3708	022346 105237 001000	.SVLAD: INCB ICNT
3709	022352 011637 001010	MOV (6),LAD
3710	022356 013777 001000 156444	MOV ICNT,ADISPLAY
3711	022364 000002	RTI
3712		
3713	022366 105237 001001	.KIT: INCB ICNT+1
3714	022372 013777 001000 156430 .OVER:	MOV ICNT,ADISPLAY
3715	022400 005737 001010	TST LAD
3716	022404 001760	BEQ .SVLAD
3717	022406 013716 001010	MOV LAD,(6)
3718	022412 000002	RTI
3719		
3720	022414 000100	TIMES: 100

3721			.SBTTL	\$HLT - HLT ROUTINE (ERROR TYPEOUT)	
3722					
3723			:	THIS ROUTINE PRINTS OUT ERROR MESSAGES STARTING WITH THE	
3724			:	ADDRESS OF THE "HLT". IT ALSO COUNTS THE NUMBER OF ERRORS	
3725			:	AND HAS THE CAPABILITY OF LOOPING ON ERROR, BELL ON ERROR,	
3726			:	"HALT" ON ERROR, AND INHIBIT TYPEOUTS. AN OPTIONAL ARGUMENT	
3727			:	(HLT+3) WILL BE PLACED IN ".HLTCT:" FOR ADDITIONAL TYPEOUTS.	
3728					
3729	022416	104416	.HLT:	KBDIN	;GO CHECK FOR IG
3730	022420	032777	002000	156400	BIT #SW10,JSWR ;BELL ON ERROR?
3731	022426	001402			BEG IS ;NO - SKIP
3732	022430	104402	000007		TYPE ,BELL ;RING BELL
3733	022434	005237	001002	15:	INC ERRORS ;COUNT THE NUMBER OF ERRORS
3734	022440	032777	020000	156360	BIT #SW13,JSWR ;SKIP TYPEOUT IF SET
3735	022446	001025			BNE Z\$;SKIP TYPEOUTS
3736	022450	104402	022454		TYPE .ASCIZ <15><12>
3737	022460	011637	001012		MOV (6),HLTADR ;PUT ADDRESS OF INSTRUCTION ON STACK
3738	022464	162737	000002	001012	SUB #2,HLTADR ;FUDGE ADDRESS
3739	022472	117737	156314	022554	MOVB @HLTADR,.HLTCT ;GET HLT ARGUMENT
3740	022500	013746	001012		MOV HLTADR,-(6) ;PUT HLTADR ON STACK
3741	022504	104404			TYPE0 ;TYPE STACK IN OCTAL
3742	022506	104402	022512		TYPE .ASCIZ "
3743	022516	004737	026066		JSR PC,RSREG ;GO TO USER ERROR ROUTINE
3744	022522	005777	156300	Z\$:	TST JSWR ;HALT ON ERROR
3745	022526	100001			BPL .+4 ;SKIP IF CONTINUE
3746	022530	000000			HALT ;HALT ON ERROR!
3747	022532	032777	001000	156266	BIT #SW9,JSWR ;CHECK FOR INHIBIT LOOP ON ERROR
3748	022540	001003			BNE Z\$;SKIP IF LOOP ON ERROR
3749	022542	105037	001001		CLRE ICNT+1 ;CLEAR ITERATION COUNT
3750	022546	000002			RTI ;RETURN
3751	022550	000137	022366	3\$:	JMP .KIT ;LOOP ON TEST UNTIL NO ERRORS
3752					
3753	022554	000000	.HLTCT: 0		;HLT ARGUMENT


```

3754          .SBTTL          SOCTAL - OCTAL TYPEOUT ROUTINE
3755
3756          ;THIS ROUTINE IS USED TO TYPE AN OCTAL NUMBER ON THE TTY. IT WILL TYPE
3757          ;ALL 6 CHARACTERS, SUPPRESS LEADING ZEROES, OR TYPE THE
3758          ;16 BITS. IT IS CALLED VIA THE TYPOCT, TYPBIT, OR TYPOCS MACRO'S.
3759
3760 022556 012737 170101 022744 .TYPEB: MOV      #170101,.PR      ;SET BIT FLAG AND 16. CHARACTER COUNT
3761 022564 000411              BR          .PTIT              ;NOW TYPE IT IN BIT FORM
3762 022566 112737 000001 022744 .TYPEO: MOVB     #1,.PR        ;SET ZERO FILL SWITCH
3763 022574 000402              BR          .+6                ;SKIP
3764 022576 005037 022744 .TYPES: CLR          .PR              ;SUPPRESS LEADING ZERO'S
3765 022602 112737 177772 022745 .MOV        #-6,.PR+1        ;SET COUNT
3766 022610
3767 022610 010446              MOV        R4,-(6)            ;PUSH R4 ON STACK
3768 022612 010546              MOV        R5,-(6)            ;PUSH R5 ON STACK
3769 022614 016605 000010      MOV        10(6),R5            ;GET THE DATA
3770 022620 012704 022746      MOV        #.PR+2,R4            ;SET POINTER TO FIRST ASCII CHAR.
3771 022624 105014              CLRB        (4)                ;CLEAR FIRST BYTE
3772 022626 000411              BR          .PRF                ;ROTATE FIRST BIT
3773 022630 105014              .PRL: CLRB        (4)            ;CLEAR BYTE OF CHARACTER
3774 022632 032737 000100 022744 .BIT        #100,.PR          ;BIT TYPING MODE?
3775 022640 001004              BNE        .PRF                ;YES - SKIP 2 ROTATES
3776 022642 006105              ROL         R5                  ;ROTATE BIT INTO C
3777 022644 106114              ROLB        (4)                ;PACK IT
3778 022646 006105              ROL         R5                  ;ROTATE BIT INTO C
3779 022650 106114              ROLB        (4)                ;PACK IT
3780 022652 006105              .PRF: ROL         R5            ;ROTATE BIT INTO C
3781 022654 106114              ROLB        (4)                ;PACK IT
3782 022656 105714              TSTB        (4)                ;IS IT ZERO?
3783 022660 001402              BEQ         .+6                ;SKIP INC
3784 022662 105237 022744      INCB         .PR                ;SET FILL SWITCH
3785 022666 105737 022744      TSTB        .PR                ;CHECK FILL SWITCH
3786 022672 001402              BEQ         .+6                ;SKIP BITSET
3787 022674 152724 000060      BISB        #'0,(4)+            ;MAKE INTO ASCII CHAR
3788 022700 105237 022745      INCB         .PR+1              ;INC COUNT
3789 022704 001351              BNE        .PRL                ;REPEAT
3790 022706 022704 022746      CMP         #.PR+2,R4            ;EMPTY BUFFER?
3791 022712 001002              BNE        .+6                ;SKIP IF NOT
3792 022714 112724 000060      MOVB        #'0,(4)+            ;LOAD 1 ZERO
3793 022720 105014              CLRB        (4)                ;NULL TERMINATOR
3794 022722 104402 022746      TYPE        .PR+2              ;TYPE IT
3795 022726 012605              MOV        (6)+,R5            ;POP STACK INTO R5
3796 022730 012604              MOV        (6)+,R4            ;POP STACK INTO R4
3797 022732 016666 000002 000004 .MOV        2(6),4(6)        ;GET RID OF
3798 022740 012616              MOV        (6)+,(6)            ;DATA WORD
3799 022742 000002              RTI                          ;RETURN
3800
3801 022744 000012          .PR:  .BLKW      12                ;COUNT, SWITCH, AND OUTPUT BUFFER

```



```

3802          .SBTTL          $POWER - POWER DOWN AND UP ROUTINES
3803
3804          ;THIS IS THE POWER FAIL ROUTINE WHICH WILL SAVE ALL
3805          ;THE GENERAL REGISTERS AND USER DEFINED REGISTERS THEN
3806          ;WAIT FOR POWER TO GO DOWN AND BE RESTORED.
3807          ;IF THERE ISN'T ENOUGH TIME FOR SAVING ALL THE REGISTERS,
3808          ;THE PROGRAM WILL HALT AT '.ILLUP'.
3809
3810 022770 012777 023114 000124 .POWER: MOV      $.ILLUP,$.PUVEC ;SET FOR FAST UP
3811 022776 012777 000340 000120      MOV      #340,$.PUVECS+2 ;PRIO:7
3812      MOV      R0,-(6) ;PUSH R0 ON STACK
3813      MOV      R1,-(6) ;PUSH R1 ON STACK
3814      MOV      R2,-(6) ;PUSH R2 ON STACK
3815      MOV      R3,-(6) ;PUSH R3 ON STACK
3816      MOV      R4,-(6) ;PUSH R4 ON STACK
3817      MOV      R5,-(6) ;PUSH R5 ON STACK
3818 023020 010637 023120      MOV      SP,$.SAVR6 ;SAVE SP
3819 023024 012777 023034 000070      MOV      $.POWUP,$.PUVEC ;SET UP VECTOR
3820 023032 000000      HALT ;WAIT FOR PF
3821
3822 023034 013706 023120      .POWUP: MOV      $.SAVR6,SP ;GET SP
3823 023040 005001      CLR      R1 ;WAIT LOOP FOR THE TTY
3824 023042 005201 1$:      INC      R1 ;WAIT FOR THE INC
3825 023044 001376      BNE      1$ ;OF WORD
3826 023046 012605      MOV      (6)+,R5 ;POP STACK INTO R5
3827 023050 012604      MOV      (6)+,R4 ;POP STACK INTO R4
3828 023052 012603      MOV      (6)+,R3 ;POP STACK INTO R3
3829 023054 012602      MOV      (6)+,R2 ;POP STACK INTO R2
3830 023056 012601      MOV      (6)+,R1 ;POP STACK INTO R1
3831 023060 012600      MOV      (6)+,R0 ;POP STACK INTO R0
3832 023062 012737 022770 000024      MOV      $.POWER,$#24 ;SET UP THE POWER DOWN VECTOR
3833 023070 012737 000340 000026      MOV      #340,$#26 ;PRIO:7
3834 023076 104402 023102      TYPE      ,.+2 ;ASCIZ <15><12>"POWER"
3835 023112 000002      RTI ;RETURN
3836
3837 023114 000000      .ILLUP: HALT ;THE POWER UP SEQUENCE WAS STARTED
3838 023116 000776      BR      .-2 ;BEFORE THE POWER DOWN WAS COMPLETE
3839
3840 023120 000000      .SAVR6: 0 ;PUT THE SP HERE
3841 023122 000024 000026      .PUVEC: 24,26 ;POWER UP VECTOR

```


ADDRESS	HEX ADDRESS	HEX DATA	HEX DATA	ASSEMBLY	COMMENT
3842				.SBTTL	
3843					\$RDOCT - OCTAL INPUT ROUTINE
3844					;THIS ROUTINE CALLS RDLIN, INPUTS A LINE FROM THE TTY AND CONVERTS
3845					;IT INTO AN OCTAL NUMBER WHICH IS THE FIRST WORD ON THE STACK.
3846					
3847	023126	011646		.RDOCT: MOV	(6),-(6) ;MOVE THE PC
3848	023130	016666	000004 000002	MOV	4(6),2(6) ;MOVE THE PS
3849	023136	010146		MOV	R1,-(6) ;PUSH R1 ON STACK
3850	023140	010246		MOV	R2,-(6) ;PUSH R2 ON STACK
3851	023142	010346		MOV	R3,-(6) ;PUSH R3 ON STACK
3852	023144	104412		4\$: RDLIN	;READ A LINE INTO INPUT
3853	023146	005001		CLR	R1 ;INIT DATA WORD
3854	023150	005037	027276	CLR	CTN ;CLEAR COUNT WORD
3855	023154	012703	023402	MOV	#INPUT,R3 ;INIT POINTER
3856	023160	112302		1\$: MOVB	(3)+,R2 ;GET A BYTE
3857	023162	122702	000015	CMPB	#15,R2 ;WAS IT A CR?
3858	023166	001421		BEQ	2\$;GET OUT IF YES
3859	023170	122702	000060	CMPB	#'0,R2 ;CHECK FOR 0 OR GREATER
3860	023174	003024		BGT	3\$;ERROR - LESS THAN 0
3861	023176	122702	000067	CMPB	#'7,R2 ;CHECK FOR 7 OR LESS
3862	023202	002421		BLT	3\$;ERROR - GREATER THAN 7
3863	023204	006002		ROR	R2 ;GET
3864	023206	006002		ROR	R2 ;INTO
3865	023210	006002		ROR	R2 ;POSITION
3866	023212	006101		ROL	R1 ;FIRST BIT
3867	023214	006102		ROL	R2 ;GET
3868	023216	006101		ROL	R1 ;SECOND BIT
3869	023220	006102		ROL	R2 ;GET
3870	023222	006101		ROL	R1 ;THIRD BIT
3871	023224	005237	027276	INC	CTN ;YES HE TYPED SOMETHING
3872	023230	000753		BR	1\$;LOOP
3873	023232	010166	000012	2\$: MOV	R1,12(6) ;SAVE THE RESULT
3874	023236	012603		MOV	(6)+,R3 ;POP STACK INTO R3
3875	023240	012602		MOV	(6)+,R2 ;POP STACK INTO R2
3876	023242	012601		MOV	(6)+,R1 ;POP STACK INTO R1
3877	023244	000002		RTI	;RETURN
3878					
3879	023246			3\$: TYPE	+2
3880	023246	104402	023252	BR	4\$;ASCIZ "?"(15)(12)
3881	023256	000732			;TRY AGAIN

G08

3882
3883
3884
3885
3886
3887
3888
3889 023260 010546
3890 023262 012705 023402
3891 023266 022705 023422
3892 023272 001423
3893 023274 105737 177560
3894 023300 100375
3895 023302 113715 177562
3896 023306 142715 000200
3897 023312 122715 000025
3898 023316 001006
3899 023320 104402 023324
3900 023332 000753
3901 023334 122715 000177
3902 023340 001005
3903 023342
3904 023342 104402 023346
3905 023352 000743
3906 023354 111527 000000
3907 023360 104402 023356
3908 023364 122725 000015
3909 023370 001336
3910 023372 104402 000012
3911 023376 012605
3912 023400 000002
3913
3914 023402 000020
3915
3916
3917
3918
3919
3920
3921
3922 023422 011646
3923 023424 162716 000002
3924 023430 017616 000000
3925 023434 062716 117042
3926 023440 013607
3927
3928 023442 022222
3929 023444 022064
3930 023446 022566
3931 023450 022576
3932 023452 023126
3933 023454 023260
3934 023456 026050
3935 023460 027126
3936 023462 027006
3937 023464 027210

.SBTTL

SRDLIN - TTY INPUT ROUTINE

;THIS ROUTINE INPUTS A LINE TERMINATED BY A RETURN INTO ADDRESS
;INPUT AND RETURNS A LINE FEED. THE BUFFER HAS A NULL TERMINATOR
;INSTEAD OF THE RETURN. RUBOUTS ARE HANDLED BY RETYPING
;THE LINE. BUFFER OVERFLOW ERRORS LIKE A RUBOUT.

.RDLIN: MOV R5, -(6) ;SAVE R5
1\$: MOV #INPUT, R5 ;GET ADDRESS
2\$: CMP #INPUT+16., R5 ;BUFFER FULL?
BEQ 4\$;YES - TYPE "?"
TSTB @#177560 ;WAIT FOR
BPL -4 ;A CHARACTER
MOVB @#177562, (5) ;GET CHARACTER
BICB #200, (5) ;GET RID OF JUNK
CMPB #25, (5) ;IS IT A TU
BNE 5\$;BRANCH IF NOT
TYPE +2 ;ASCIZ "TU" (15) (12)
BR 1\$;START OVER
5\$: CMPB #177, (5) ;IS IT A RUBOUT
BNE 3\$;SKIP IF NOT
4\$: TYPE +2 ;ASCIZ "?" (15) (12)
BR 1\$;ZAP THE BUFFER AND LOOP
3\$: MOVB (5), #0 ;SET UP FOR TYPING
TYPE 3\$+2 ;ECHO IT
CMPB #15, (5)+ ;CHECK FOR RETURN
BNE 2\$;LOOP IF NOT RETURN
TYPE 12 ;TYPE A LINE FEED
MOV (6)+, R5 ;RESTORE R5
RTI ;RETURN

INPUT: .BLKB 16. ;TTY INPUT AREA
.SBTTL STRAP - TRAP HANDLER

;THIS ROUTINE DECODES A TRAP CALL AND JUMPS TO THE APROPRATE
;SUBROUTINE. THE CALL IS A "TRAP+N" WHERE N IS A MULTIPLE OF 2.
;THE "SET" MACRO WILL CREATE THE TABLE NEEDED. IT HAS TO
;FOLLOW THIS MACRO.

.TRAP: MOV (6), -(6) ;GET ADDRESS OF TRAP +2
SUB #2, (6) ;MAKE IT ADDRESS OF TRAP
MOV @ (6), (6) ;GET TRAP INSTRUCTION
ADD #.TRP+2-TRAP, (6) ;GET DATA AND MAKE IT AN OFFSET
.TRP: MOV @ (6)+, PC ;GO TO PROPER SUBROUTINE

.SCOPE	=	TRAP+0	(104400)
.TYPE	=	TRAP+2	(104402)
.TYPE0	=	TRAP+4	(104404)
.TYPES	=	TRAP+6	(104406)
.RDOCT	=	TRAP+10	(104410)
.RDLIN	=	TRAP+12	(104412)
.CLRDK	=	TRAP+14	(104414)
.KBDIN	=	TRAP+16	(104416)
.SUSWR	=	TRAP+20	(104420)
.CNTLU	=	TRAP+22	(104422)


```

3938      ;ROUTINE TO ALLOW THE OPERATOR TO SET BITS
3939      ;IN THE I/O REGISTERS VIA THE SWITCH REGISTER
3940
3941      ;WORD COUNT REGISTER
3942      SRSWC: MOV      #500,SP      ;SET UP STACK FOR TRAP CALL
3943      1$:   KBDIN                      ;CHECK THE WORLD
3944      MOV      @SWR,@RSWC      ;MOV SWR INTO WORD COUNT REG
3945      MOV      @RSWC,@DISPLAY  ;DISPLAY IN LIGHTS
3946      BR       1$
3947
3948      ;CURRENT ADDRESS REGISTER
3949      SRSBA: MOV      #500,SP      ;INIT THE STACK
3950      1$:   KBDIN                      ;CTW
3951      MOV      @SWR,@RSBA      ;MOV SWR INTO CURRENT ADDR REG
3952      MOV      @RSBA,@DISPLAY  ;SHOW IN LIGHTS
3953      BR       1$
3954
3955      ;DISK ADDRESS REGISTER
3956      SRSDA: MOV      #500,SP      ;INIT THE STACK
3957      1$:   KBDIN                      ;CTW
3958      MOV      @SWR,@RSDA      ;MOV SWR INTO DISK ADDR REG
3959      MOV      @RSDA,@DISPLAY  ;SHOW IN LIGHTS
3960      BR       1$
3961
3962      ;DRIVE STATUS REGISTER
3963      SRSDS: MOV      #500,SP      ;INIT THE STACK
3964      1$:   KBDIN                      ;CTW
3965      MOV      @SWR,@RSDS      ;MOV SWR INTO DRIVE STATUS
3966      MOV      @RSDS,@DISPLAY  ;SHOW IN LIGHTS
3967      BR       1$
3968
3969      ;DRIVE ERROR REGISTER
3970      SRSER: MOV      #500,SP      ;INIT THE STACK
3971      1$:   KBDIN                      ;CTW
3972      MOV      @SWR,@RSER      ;LOAD ER REG
3973      MOV      @RSER,@DISPLAY  ;DISPLAY IT IN LIGHTS
3974      BR       1$
3975      ;LOOP
3976
3977      ;WATCH LOOK AHEAD REGISTER
3978      SRSLA: MOV      @RSLA,@DISPLAY ;SHOW IN LIGHTS
              BR       SRSLA

```



```

3979                                     :RSCS2 REGISTER
3980 023642 012706 000500 SRCS2: MOV #500,SP ;INIT THE STACK
3981 023646 104416 1$: KBDIN ;CTW
3982 023650 017777 155152 155230 MOV @SWR,@RSCS2 ;LOAD CS2
3983 023656 017777 155224 155144 MOV @RSCS2,@DISPLAY ;DISPLAY IT
3984 023664 000770 BR 1$
3985
3986                                     :RSAS REGISTER
3987 023666 012706 000500 SRAS: MOV #500,SP ;INIT THE STACK
3988 023672 104416 1$: KBDIN ;CTW
3989 023674 017777 155126 155220 MOV @SWR,@RSAS ;LOAD RSAS
3990 023702 017777 155214 155120 MOV @RSAS,@DISPLAY ;DISPLAY IT
3991 023710 000770 BR 1$
3992
3993                                     :RSMR REGISTER
3994 023712 012706 000500 RSMR: MOV #500,SP ;INIT THE STACK
3995 023716 104416 1$: KBDIN ;CTW
3996 023720 017777 155102 155202 MOV @SWR,@RSMR ;LOAD RSMR
3997 023726 017777 155176 155074 MOV @RSMR,@DISPLAY ;DISPLAY IT
3998 023734 000770 BR 1$
3999
4000                                     :DISK CONTROL STATUS REGISTER
4001 023736 012706 000500 SRSCS1: MOV #500,SP ;INIT THE STACK
4002 023742 104416 1$: KBDIN ;CTW
4003 023744 012737 000340 177776 MOV #340,@PS ;LOCK UP INTERRUPTS
4004 023752 012777 177777 155130 MOV #177777,@RSCS1 ;SET WORD COUNT -1 WORD
4005 023760 013777 001102 155124 MOV @#0BUFSV,@RSBA ;SET UP CURRENT ADDRESS
4006 023766 017777 155034 155110 MOV @SWR,@RSCS1 ;MOV SWR INTO CONTROL REG
4007 023774 032777 000001 155102 BIT #BIT0,@RSCS1 ;IS FUNCTION BITS SET
4008 024002 001757 BEQ 1$ ;FUNCTION BITS NOT SET
4009 024004 105777 155074 2$: TSTB @RSCS1 ;TEST FOR DISK READY
4010 024010 100375 BPL 2$ ;DISK STILL NOT READY
4011 024012 000753 BR 1$ ;DISK NOT BUSY SECT NEW CR

```



```

4012      ;THIS ROUTINE GIVES THE OPERATOR THE ABILITY TO
4013      ;SELECT DA, WC, UNIT # AND DESIRED PATTERN. PATTERN = NUMBER TYPED
4014      ;WITH SW12 SET THE PROGRAM WILL LOOP ON A READ WITH LOC OUTBUF+2 AS
4015      ;THE BA ADDR. WITH BIT12 0 IN THE SWR THE PROGRAM
4016      ;WILL WRITE WITH OUTBUF AS THE BA ADDR. BAI IS ALWAYS SET
4017      ;SWITCHES 0 TO 11 WILL DETERMINE THE DA
4018      ;EXAMPLE:      TYPE UNIT # 5
4019      ;                  TYPE POSITIVE (OCTAL) WC 64
4020      ;                  TYPE PATTERN DESIRED 1252525
4021
4022      024014 012706 000500      TKSEL:  MOV      #500,SP          ;SET STACK
4023      024020 000240      NOP
4024      024022 104402 024026      TYPE      ,.+2              ;.ASCIZ <15><12>"TYPE UNIT # "
4025      024046 104410      RDOCT
4026      024050 012637 001152      MOV      (6)+,UNNUM
4027      024054 104402 024060      TYPE      ,.+2              ;.ASCIZ <15><12>"TYPE POSITIVE (OCTAL) WC "
4028      024114 104410      RDOCT
4029      024116 012637 001174      MOV      (6)+,WORK
4030      024122 005137 001174      COM      WORK
4031      024126 104402 024132      TYPE      ,.+2              ;.ASCIZ <15><12>"TYPE PATTERN DESIRED "
4032      024162 104410      RDOCT
4033      024164 012637 027364      MOV      (6)+,OUTBUF
4034      024170 042737 000001 027124      BIC      #BIT0,SWI          ;CLEAR THE BEENHEREBIT
4035      024176 104420      SUSWR          ;INIT SWITCHLESS
4036      024200 017737 154622 001200      TK1:  MOV      @SWR,WORK2      ;SAVE SWR
4037      024206 104414      TK2:  CLRDK          ;CLEAR ALL RS REG
4038      024210 052777 000010 154670      BIS      #BIT3,@RSCS2      ;SET BAI
4039      024216 017737 154604 001176      MOV      @SWR,WORK1      ;GET SWR FOR DSK ADDR
4040      024224 042737 170000 001176      BIC      #170000,WORK1      ;CLEAR UNIT #
4041      024232 013777 001176 154654      TKKS:  MOV      WORK1,@RSDA      ;LOAD THE DA
4042      024240 013777 001174 154642      MOV      WORK,@RSMC      ;LOAD WORD COUNT
4043      024246 032777 010000 154552      BIT      #BIT12,@SWR      ;READ?
4044      024254 001412      BEQ      WTE          ;NO
4045      024256 012777 027366 154626      MOV      #OUTBUF+2,@RSBA      ;LOAD CURRENT ADDRESS
4046      024264 012777 000071 154612      MOV      #71,@RSCS1      ;GO AND READ
4047      024272 105777 154606      WT:  TSTB      @RSCS1          ;TEST FOR READY
4048      024276 100375      BPL      .-4
4049      024300 000407      BR      SWRCHG
4050      024302 012777 027364 154602      WTE:  MOV      #OUTBUF,@RSBA
4051      024310 012777 000061 154566      MOV      #61,@RSCS1
4052      024316 000765      BR      WT
4053      024320 017777 154560 154502      SWRCHG: MOV      @RSCS1,@DISPLAY ;DISPLAY CS1
4054      024326 005777 154552      TST      @RSCS1          ;ANY ERRORS?
4055      024332 100001      BPL      IS          ;NO
4056      024334 104014      HLT      !DA!WC
4057      024336 104416      IS:  KBDIN          ;CHECK FOR NEW VALUE
4058      024340 027737 154462 001200      CMP      @SWR,WORK2      ;DID SWR CHANGE?
4059      024346 001314      BNE      TK1          ;YES
4060      024350 000716      BR      TK2          ;NO

```


K08

MAINDEC-11-DZRSB-E
DZRSBE.P11

TST122

RH11-R503LA-R503-R504 BASIC FUNCTION DIAGNOSTIC
WRITE LOCK TEST

MACY11 27(732) 25-SEP-76 09:06 PAGE 102

```

4061 ;*****
4062 ;TEST 122 WRITE LOCK TEST
4063 ;*****
4064 024352 104400 TST122: SCOPE
4065
4066 WRTLCK:
4067 024354 104402 024360 TYPE ;.ASCIZ <15><12>"LOAD SW WITH UNIT # AND CONT"
4068 024420 022737 000176 001026 CMP #SWREG,SWR ;CHECK IF SWITCHLESS CPU
4069 024426 001402 BEQ 7$
4070 024430 000000 HALT
4071 024432 000401 BR 8$ ;GO AROUND TRAP CALL
4072 024434 104422 7$: CNTLU ;GET SWREG VALUE
4073 024436 017737 154364 001152 8$: MOV #SWR,UNNUM ;GET UNIT #
4074 024444 104414 CLRDK ;CLEAR ALL REG
4075 024446 005037 027364 CLR OUTBUF ;PUT A 0 INTO DATA BUFFER
4076 024452 012777 027364 154432 MOV #OUTBUF,ARSDA ;SETUP REG TO
4077 024460 012777 007700 154426 MOV #7700,ARSDA ;TO A WRITE
4078 024466 012777 177777 154414 MOV #-1,ARSWC
4079 024474 012777 000061 154402 MOV #61,ARSCS1
4080 024502 105777 154376 6$: TSTB ARSCS1 ;WAIT FOR DONE
4081 024506 100375 BPL 6$
4082 024510 104402 024514 TYPE ;.ASCIZ <15><12>"SET WRITE LOCK ENABLE AND CONTINUE"
4083 024562 000000 HALT ;. +2
4084 024564 012777 000000 154322 1$: MOV #0,ARSDA
4085 024572 022777 014600 154316 CMP #14600,ARSDS
4086 024600 001401 BEQ ;+4
4087 024602 104044 HLT !DS!DA ;DS SHOULD=14600
4088 024604 012777 000100 154302 MOV #100,ARSDA
4089 024612 022777 010600 154276 CMP #10600,ARSDS
4090 024620 001401 BEQ ;+4
4091 024622 104044 HLT !DS!DA ;DS SHOULD=10600
4092 024624 104402 024630 TYPE ;.ASCIZ <15><12>"SET WRT LOC SW 0 AND CONT"
4093 024664 000000 HALT ;. +2
4094 024666 022777 014600 154222 2$: CMP #14600,ARSDS
4095 024674 001401 BEQ ;+4
4096 024676 104044 HLT !DS!DA ;DS SHOULD=14600
4097 024700 012777 000300 154206 MOV #300,ARSDA
4098 024706 022777 010600 154202 CMP #10600,ARSDS
4099 024714 001401 BEQ ;+4
4100 024716 104044 HLT !DS!DA ;DS SHOULD=10600
4101 024720 104402 024724 TYPE ;.ASCIZ <15><12>"SET WRT LOC SW 1 AND CONT"
4102 024760 000000 HALT ;. +2
4103 024762 022777 014600 154126 3$: CMP #14600,ARSDS
4104 024770 001401 BEQ ;+4
4105 024772 104044 HLT !DS!DA ;DS SHOULD=14600
4106 024774 012777 000700 154112 MOV #700,ARSDA
4107 025002 022777 010600 154106 CMP #10600,ARSDS
4108 025010 001401 BEQ ;+4
4109 025012 104044 HLT !DS!DA ;DS SHOULD=10600
4110 025014 104402 025020 TYPE ;.ASCIZ <15><12>"SET WRT LCK SW 2 AND CONT"
4111 025054 000000 HALT ;. +2

```


4112	025056	022777	014600	154032	4\$:	CMP	#14600, ARSDS	
4113	025064	001401				BEQ	.+4	
4114	025066	104044				HLT	!DS!DA	;DS SHOULD=14600
4115	025070	012777	001700	154016		MOV	#1700, ARSDA	
4116	025076	022777	010600	154012		CMP	#10600, ARSDS	
4117	025104	001401				BEQ	.+4	
4118	025106	104044				HLT	!DS!DA	;DS SHOULD=10600
4119	025110	104402	025114			TYPE	,.+2	;.ASCIZ <15><12>"SET WRT LCK SW 3 AND CONT"
4120	025150	000000			HALT:	HALT		
4121	025152	022777	014600	153736		CMP	#14600, ARSDS	
4122	025160	001401				BEQ	.+4	
4123	025162	104044				HLT	!DS!DA	;DS SHOULD=14600
4124	025164	012777	003700	153722		MOV	#3700, ARSDA	
4125	025172	022777	010600	153716		CMP	#10600, ARSDS	
4126	025200	001401				BEQ	.+4	
4127	025202	104044				HLT	!DS!DA	;DS SHOULD=10600
4128	025204	104402	025210			TYPE	,.+2	;.ASCIZ <15><12>"SET WRT LCK SW 4 AND CONT"
4129	025244	000000				HALT		
4130	025246	022777	014600	153642		CMP	#14600, ARSDS	
4131	025254	001401				BEQ	.+4	
4132	025256	104044				HLT	!DS!DA	;DS SHOULD=14600
4133	025260	012777	006000	153626		MOV	#6000, ARSDA	
4134	025266	022777	010600	153622		CMP	#10600, ARSDS	
4135	025274	001401				BEQ	.+4	
4136	025276	104044				HLT	!DS!DA	;DS SHOULD=10600
4137	025300	104402	025304			TYPE	,.+2	;.ASCIZ <15><12>"SET WRT LCK SW 5 AND CONT"
4138	025340	000000				HALT		
4139	025342	022777	014600	153546		CMP	#14600, ARSDS	
4140	025350	001401				BEQ	.+4	
4141	025352	104044				HLT	!DS!DA	;DS SHOULD=14600
4142	025354	012777	007700	153532		MOV	#7700, ARSDA	
4143	025362	022777	014600	153526		CMP	#14600, ARSDS	
4144	025370	001401				BEQ	.+4	
4145	025372	104044				HLT	!DS!DA	;DS SHOULD=14600
4146	025374	012737	177777	027364		MOV	#-1, OUTBUF	;PUT A 1 INTO DATA BUFFER
4147	025402	012777	027364	153502		MOV	#OUTBUF, ARSBA	;SETUP REG TO
4148	025410	012777	007700	153476		MOV	#7700, ARSDA	;TO A WRITE
4149	025416	012777	177777	153464		MOV	#-1, ARSWC	
4150	025424	012777	000061	153452		MOV	#61, ARSCS1	;TRY TO WRITE
4151	025432	105777	153446		7\$:	TSTB	ARSCS1	;WAIT FOR DONE
4152	025436	100375				BPL	7\$	
4153	025440	105777	153440		5\$:	TSTB	ARSCS1	;WAIT FOR READY
4154	025444	100375				BPL	5\$	
4155	025446	022777	154600	153442		CMP	#154600, ARSDS	
4156	025454	001401				BEQ	.+4	
4157	025456	104044				HLT	!DS!DA	;DS SHOULD=154600

M08

MAINDEC-11-DZRSB-E RH11-RS03LA-RS03-RS04 BASIC FUNCTION DIAGNOSTIC MACY11 27(732) 25-SEP-76 09:06 PAGE 104
 DZRSBE.P11 TST122 WRITE LOCK TEST

4158	025460	022777	004000	153432		CMP	#4000, ARSER	
4159	025466	001401				BEQ	+4	
4160	025470	104044				HLT	!DS!DA	;ER SHOULD=4000
4161	025472	104414				CLRDK		;CLEAR ALL REG
4162	025474	022777	014600	153414		CMP	#14600, ARSDS	;DID WLE CLEAR?
4163	025502	001401				BEQ	+4	;NO
4164	025504	104040				HLT	!DS	;A CLEAR SHOULD NOT CLEAR WLE
4165	025506	005037	027364			CLR	OUTBUF	;PUT A 0 INTO DATA BUFFER
4166	025512	012777	027364	153372		MOV	#OUTBUF, ARSBA	;SETUP REG TO
4167	025520	012777	007700	153366		MOV	#7700, ARSDA	;TO A WRITE
4168	025526	012777	177777	153354		MOV	#-1, ARSWC	
4169	025534	012777	000071	153342		MOV	#71, ARSCS1	;DO A READ TO SEE IF DISK DID
4170	025542	105777	153336		BS:	TSTB	ARSCS1	;ACTUALLY GET WRITTEN ON TO
4171	025546	100375				BPL	BS	;WAIT FOR DONE
4172	025550	005737	027364			TST	OUTBUF	;IS DATA STILL 0 ON THE DSK?
4173	025554	001401				BEQ	+4	;YES
4174	025556	104040				HLT	!DS	;NO DSK DID GET WRITTEN ONTO WITH WLE SET
4175	025560	104402	025564			TYPE	,+2	;ASCIZ <15><12>"RESET ALL W/L SWITCHES"
4176	025616	000000				HALT		
4177	025620	022777	010600	153270		CMP	#10600, ARSDS	;DID WLE CLEAR
4178	025626	001401				BEQ	+4	;YES
4179	025630	104040				HLT	!DS	;NO
4180								
4181								
4182	025632	012737	025716	000004		ROUTINE FOR FINDING MEMORY ON "B" PORT		
4183	025640	012737	000340	000006	FINDM:	MOV	#MAXREF, 4	;SET UP I/O BUS TRAP
4184	025646	032777	010000	153152		MOV	#340, 6	;SET PS
4185	025654	001003				BIT	#BIT12, ASWR	;INHIBIT OBUFSV FROM CHANGING?
4186	025656	062737	020000	001102		BNE	EXREF	;YES
4187	025664	005777	153212		EXREF:	ADD	#20000, OBUFSV	;ADD 4 K
4188	025670	012737	000006	000004		TST	OBUSV	;LEGAL LOC ? IF NO TRAPS
4189	025676	005037	000006			MOV	#6, 4	;RESTORE I/O BUS TRAP
4190	025702	022737	177446	001102		CLR	6	
4191	025710	103403				CMP	#177446, OBUFSV	;TEST FOR GREATER THEN 28K
4192	025712	000137	006736			BLO	MAXRF1	;YES
4193	025716	022626				JMP	WRTSTB	;RETRY WRITING
4194	025720	012737	000006	000004	MAXREF:	CMP	(6)+, (6)+	;CLEAR STACK
4195	025726	005037	000006		MAXRF1:	MOV	#6, 4	;RESTORE I/O BUS TRAP
4196	025732	012737	027364	001102		CLR	6	
4197	025740	032777	020000	153060		MOV	#OUTBUF, OBUFSV	;RESTORE ORIGINAL VALUE
4198	025746	001031				BIT	#BIT13, ASWR	;INHIBIT TYPEOUT?
4199	025750	032737	020000	001160		BNE	1\$	;YES
4200	025756	001025				BIT	#BIT13, ONCEE	;DID WE TYPE THIS YET?
4201	025760	104402	025764			BNE	1\$	;YES
4202						TYPE	,+2	;ASCIZ <15><12>"COULD NOT FIND MEMORY ON -B- PORT"<15><
4203	026032	052737	020000	001160	EVEN			
4204	026040	104400			1\$:	BIS	#BIT13, ONCEE	;SET TYPE FLAG
4205	026042	104400				SCOPE		;UPDATE
4206	026044	000137	007526			SCOPE		;TEST NUMBERS
4207						JMP	NXM	;CONT TESTING
4208								
4209	026050	012777	000040	153030		CLEAR ALL DISK REGISTERS		
4210	026056	013777	001152	153022	.CLRDK:	MOV	#40, ARSCS2	;CLEAR ALL DSK REG
4211	026064	000002				MOV	UNNUM, ARSCS2	;GET UNIT NUMBER
						RTI		


```

4212                                     ;ERROR TYPTXTOUT ROUTINE
4213
4214 026066 005737 022554      RSREG:  TST      .HLTCT      ;SHOULD WE TYPTXT GOOD AND BAD
4215 026072 001021                                     ;NO
4216 026074 104402 026100      TYPE      +2      ;ASCIZ " BAD="
4217 026106 010046                                     ;PUT BAD ON STACK
4218 026110 104404      TYPE0      ;TYPE STACK IN OCTAL
4219 026112 104402 026116      TYPE      +2      ;ASCIZ " GOOD="
4220 026126 010146      MOV      GOOD,-(6)      ;PUT GOOD ON STACK
4221 026130 104404      TYPE0      ;TYPE STACK IN OCTAL
4222 026132 000137 026570      JMP      PTDONE      ;GET OUT
4223 026136
4224 026136 104402 026142      8$:      TYPE      +2      ;ASCIZ " CS1="
4225 026150 017746 152730      MOV      @RSCS1,-(6)      ;PUT @RSCS1 ON STACK
4226 026154 104404      TYPE0      ;TYPE STACK IN OCTAL
4227 026156
4228 026156 104402 026162      1$:      TYPE      +2      ;ASCIZ " FR="
4229 026170 017746 152724      MOV      @RSEF,-(6)      ;PUT @RSEF ON STACK
4230 026174 104404      TYPE0      ;TYPE STACK IN OCTAL
4231 026176
4232 026176 104402 026202      2$:      TYPE      +2      ;ASCIZ " CS2="
4233 026210 017746 152672      MOV      @RSCS2,-(6)      ;PUT @RSCS2 ON STACK
4234 026214 104404      TYPE0      ;TYPE STACK IN OCTAL
4235 026216 032737 000200 022554      BIT      #200,.HLTCT      ;TYPTXT SECOND SET ?
4236 026224 001076      BNE      SEEC      ;YES
4237 026226 032737 000100 022554      BIT      #AS,.HLTCT      ;TYPTXT ER ?
4238 026234 001410      BEQ      3$      ;NO
4239 026236 104402 026242      TYPE      +2      ;ASCIZ " AS="
4240 026250 017746 152646      MOV      @RSAS,-(6)      ;PUT @RSAS ON STACK
4241 026254 104404      TYPE0      ;TYPE STACK IN OCTAL
4242 026256 032737 000020 022554      3$:      BIT      #BA,.HLTCT      ;TYPTXT BUS ASSRESS
4243 026264 001410      BEQ      4$      ;NO
4244 026266 104402 026272      TYPE      +2      ;ASCIZ " BA="
4245 026300 017746 152606      MOV      @RSBA,-(6)      ;PUT @RSBA ON STACK
4246 026304 104404      TYPE0      ;TYPE STACK IN OCTAL
4247 026306 032737 000004 022554      4$:      BIT      #DA,.HLTCT      ;TYPTXT DA ?
4248 026314 001410      BEQ      5$      ;NO
4249 026316 104402 026322      TYPE      +2      ;ASCIZ " DA="
4250 026330 017746 152560      MOV      @RSDA,-(6)      ;PUT @RSDA ON STACK
4251 026334 104404      TYPE0      ;TYPE STACK IN OCTAL
4252 026336 032737 000010 022554      5$:      BIT      #WC,.HLTCT      ;TYPTXT WC?
4253 026344 001410      BEQ      6$      ;NO
4254 026346 104402 026352      TYPE      +2      ;ASCIZ " WC="
4255 026360 017746 152524      MOV      @RSWC,-(6)      ;PUT @RSWC ON STACK
4256 026364 104404      TYPE0      ;TYPE STACK IN OCTAL

```


4257	026366	032737	000040	022554	6\$:	BIT	#DS,.HLTCT	;DRIVE STATUS
4258	026374	001475				BEQ	PTDONE	;NO
4259	026376	104402	026402			TYPE	.+2	;ASCIZ "DS="
4260	026410	017746	152502			MOV	ARSDS,-(6)	;PUT ARSDS ON STACK
4261	026414	104404				TYPE0		;TYPE STACK IN OCTAL
4262	026416	000137	026570			JMP	PTDONE	;GET OUT
4263	026422	042737	000200	022554	SEEC:	BIC	#200,.HLTCT	;CLEAR COMMON BIT
4264	026430	032737	000240	022554		BIT	#DT,.HLTCT	;TYPTXT DRIVE TYPE?
4265	026436	001410				BEQ	9\$	;NO
4266	026440	104402	026444			TYPE	.+2	;ASCIZ "DT="
4267	026452	017746	152454			MOV	ARSDT,-(6)	;PUT ARSDT ON STACK
4268	026456	104404				TYPE0		;TYPE STACK IN OCTAL
4269	026460	032737	000210	022554	9\$:	BIT	#DB,.HLTCT	;TYPTXT DATA BUFFER
4270	026466	001410				BEQ	10\$	;NO
4271	026470	104402	026474			TYPE	.+2	;ASCIZ "DB="
4272	026502	017746	152420			MOV	ARSDB,-(6)	;PUT ARSDB ON STACK
4273	026506	104404				TYPE0		;TYPE STACK IN OCTAL
4274	026510	032737	000220	022554	10\$:	BIT	#MR,.HLTCT	;TYPTXT MN?
4275	026516	001410				BEQ	11\$	;NO
4276	026520	104402	026524			TYPE	.+2	;ASCIZ "MR="
4277	026532	017746	152372			MOV	ARSMR,-(6)	;PUT ARSMR ON STACK
4278	026536	104404				TYPE0		;TYPE STACK IN OCTAL
4279	026540	032737	000204	022554	11\$:	BIT	#LA,.HLTCT	;TYPTXT LA?
4280	026546	001410				BEQ	PTDONE	;NO
4281	026550	104402	026554			TYPE	.+2	;ASCIZ "LA="
4282	026562	017746	152336			MOV	ARSLA,-(6)	;PUT ARSLA ON STACK
4283	026566	104404				TYPE0		;TYPE STACK IN OCTAL
4284	026570	052737	100000	001160	PTDONE:	BIS	#BIT15,ONCE	;SET FORND ERROR FLAG
4285	026576	000207				RTS	PC	
4286								
4287	026600	005037	001174		WAITRY:	CLR	WORK	;CLEAR COUNTER
4288	026604	105777	152274		1\$:	TSTB	ARSCS1	;TEST READY
4289	026610	100406				BMI	2\$	;OK CONT
4290	026612	005237	001174			INC	WORK	;UPDATE COUNTER
4291	026616	005737	001174			TST	WORK	;DONE YET?
4292	026622	001403				BEQ	3\$	;READY DID NOT COME UP
4293	026624	000767				BR	1\$	;CONTINUE WAITING
4294	026626	062716	000002		2\$:	ADD	#2,(SP)	;UPDATE RETURN PC
4295	026632	000207			3\$:	RTS	PC	;RETURN

;RANDOM DATA GENERATOR SUBROUTINE

```

4296
4297
4298 026634 013737 026776 027002 RANDOM: MOV LONUM,LOSAY
4299 026642 013737 027000 027004 RAND1: MOV HINUM,HISAY
4300 026650 013700 026776 MOV LONUM,R0
4301 026654 013704 027000 MOV HINUM,R4
4302 026660 012703 000007 MOV #7,R3
4303 026664 005002 CLR R2
4304 026666 006300 SHIFT: ASL R0
4305 026670 006104 ROL R4
4306 026672 006102 ROL R2
4307 026674 005303 DEC R3
4308 026676 001373 BNE SHIFT
4309 026700 063700 026776 ADD LONUM,R0
4310 026704 005504 ADC R4
4311 026706 063704 027000 ADD HINUM,R4
4312 026712 005502 ADC R2
4313 026714 062700 001057 ADD #1057,R0
4314 026720 005504 ADC R4
4315 026722 005502 ADC R2
4316 026724 062704 047401 ADD #47401,R4
4317 026730 005502 ADC R2
4318 026732 062702 000006 ADD #6,R2
4319 026736 062700 000002 ADD #2,R0
4320 026742 005504 ADC R4
4321 026744 010037 026776 MOV R0,LONUM
4322 026750 010021 MOV R0,(R1)+
4323 026752 005337 001174 DEC WORK
4324 026756 001406 BEQ EXGEN
4325 026760 010437 027000 MOV R4,HINUM
4326 026764 010421 MOV R4,(1)+
4327 026766 005337 001174 DEC WORK
4328 026772 001326 BNE RAND1
4329 026774 000205 EXGEN: RTS
4330 026776 000000 LONUM: 0
4331 027000 000000 HINUM: 0
4332 027002 000000 LOSAV: 0
4333 027004 000000 HISAV: 0
4334 027006 RANEND:
4335
4336 027006 032737 000001 027124 .SUSMR: BIT #BIT0,SWI
4337 027014 001037 BNE XXX
4338 027016 013746 000006 MOV 6,-(SP)
4339 027022 013746 000004 MOV 4,-(SP)
4340 027026 012737 027046 000004 MOV #1$4
4341 027034 022777 177777 151764 CMP #-1,$SWR
4342 027042 001402 BEQ 2$
4343 027044 000407 BR 3$
4344 027046 022626 1$: CMP (SP)+,(SP)+
4345 027050 012737 000176 001026 2$: MOV #SWREG,SWR
4346 027056 012737 000174 001030 3$: MOV #DISPREG,DISPLAY
4347 027064 022737 000176 001026 CMP #SWREG,SWR
4348 027072 001004 BNE 4$
4349 027074 005737 000042 TST 4$
4350 027100 001001 BNE 4$
4351 027102 104422 CNTLU

```

```

;SET UP R0 WITH 5 DIGITS LOW
;SET UP R1 WITH 5 DIGITS HIGH
;SET UP SHIFT COUNT
;CLEAR R2
;SHIFT R0 LEFT AND
;ROTATE CARRY INTO LSB OF R1 INTO
;ROTATE CARRY OUT OF R1 INTO R2
;DECREMENT R3
;CONTINUE SHIFT LOOP
;ADDN IN NUMBER TO MAKE X 129
;PROPOGATE CARRY
;ADDN IN NUMBER TO MAKE X 129
;PROPOGATE CARRY
;ADDN LOW CONSTANT
;PROPOGATE CARRIES
;PROPOGATE AGAIN
;ADDN HIGH CONSTANT
;PROPOGATE CARRY
;ADDN HIGHEST CONSTANT
;REPRIME R0 WITH HIGH DIGIT
;PROPOGATE CARRY
;PUT R0 BACK IN LONUM
;LOAD WC

```

```

;PUT R1 BACK IN HINUM
;HOLD HINUM FOR PROGRAM

```

```

;RETURN TO PROGRAM

```

```

;SAVE 6 ON STACK
;SAVE 4 ON STACK
;SET UP TRAP ADDRESS
;TEST 177570
;FAKE OUT
;HARDWARE AVAILABLE
;ADJUST STACK
;SET UP SOFTWARE REGISTERS
;1ST TIME THRU?
;NO CHANGE STILL 177570
;ANY XXDP OR ACT
;SWR=000000
;GET INITIAL SETTINGS

```


4352	027104	012637	000004	4\$:	MOV	(SP)+,4	:REPLACE 4 FROM STACK
4353	027110	012637	000006		MOV	(SP)+,6	:REPLACE 6 FROM STACK
4354	027114	052737	000001	027124	XXX:	BIS	:SET THE BEENHEREBIT
4355	027122	000002			RTI		:ALL DONE
4356							
4357	027124	000000			SWI:	0	
4358							
4359							
4360							
4361	027126	005737	000042		.KBDIN:	TST	42
4362	027132	001057				BNE	OKT
4363	027134	022737	000176	001026		CMP	#SWREG,SWR
4364	027142	001053				BNE	OKT
4365	027144	105777	151650			TSTB	ATKS
4366	027150	100050				BPL	OKT
4367	027152	017737	151644	027274		MOV	ATKB,.MSG
4368	027160	042737	177600	027274		BIC	#177600,.MSG
4369	027166	122737	000007	027274		CMPB	#7,.MSG
4370	027174	001036				BNE	OKT
4371	027176	104402	027202			TYPE	..+2
4372	027210					.CNTLU:	
4373	027210	104402	027214			TYPE	..+2
4374	027224	013746	000176			MOV	SWREG,-(6)
4375	027230	104404				TYPE0	
4376	027232	104402	027236			TYPE	..+2
4377	027250	104410				RDOCT	
4378	027252	012637	027274			MOV	(SP)+,.MSG
4379	027256	005737	027276			TST	CTN
4380	027262	001403				BEG	OKT
4381	027264	013737	027274	000176		MOV	.MSG,SWREG
4382	027272	000002			OKT:	RTI	
4383							
4384	027274	000000			.MSG:	0	
4385	027276	000000			CTN:	0	

4386
4387
4388
4389
4390
4391
4392
4393
4394
4395
4396
4397
4398
4399
4400
4401
4402
4403
4404
4405
4406
4407
4408
4409
4410
4411
4412
4413
4414
4415
4416
4417
4418

027300 000003
027302 000005
027304 000007
027306 000013
027310 000015
027312 000017
027314 000023
027316 000025
027320 000027
027322 000033
027324 000035
027326 000037
027330 000041
027332 000043
027334 000045
027336 000047
027340 000000

027342 000053
027344 000055
027346 000057
027350 000063
027352 000065
027354 000073
027356 000075
027360 000077
027362 000000
027364 000300
030164 000300
000001

;TABLES FOR ILLEGAL FUNCTION TESTS

ILLTAB: 3
5
7
13
15
17
23
25
27
33
35
37
41
43
45
47
0

ILFTB2: 53
55
57
63
65
73
75
77
0

OUTBUF: .BLKW 300
INBUF: .BLKW 300
.END

M3INDEC-11-DZRSB-E RH11-RS03LA-RS03-RS04 BASIC FUNCTION DIAGNOSTIC MACY11 27(732) 25-SEP-76 09:06 PAGE 111
 DZRSBE.P11 CROSS REFERENCE TABLE -- USER SYMBOLS

B-E RH11-R503LA-R503-R504 BASIC F
CROSS REFERENCE TABLE -- USER SYMBOLS

[illegible]

CNTLU = 104422	3937#	4072	4351											
CSTST = 012050	2345#													
CS1 = 000001	737#	919	977	1017	1021	1025	1033	1303	1307	1362	1366	1492	1524	
	1555	1590	1645	1672	1674	1680	1716	1718	1724	1760	1762	1804	1806	
	1817	1848	1850	1856	1892	1894	1903	1941	1944	1981	2014	2057	2079	
	2116	2119	2134	2137	2140	2184	2187	2190	2201	2204	2216	2219	2481	
	2497	2558	2574	2598	2601	2613	2642	2676	2701	2732	2787	2853	2881	
	2950	2959	2985	2994	3029	3210	3219	3407	3413	3432	3462			
CS2 = 000200	744#	911	973	1043	1053	1057	1062	1359	1371	1388	1413	1424	1436	
	2021	2622	2915	2922										
CTN = 027276	3854#	3871#	4379	4385#										
DA = 000004	739#	930	1145	1149	1153	1174	1501	1533	1564	1599	1634	1677	1680	
	1686	1721	1771	1774	1814	1823	1853	1906	1944	1947	1953	2140	2190	
	2648	2801	2856	3049	3148	3184	3321	3568	3574	3578	4056	4087	4091	
	4096	4100	4105	4109	4114	4118	4123	4127	4132	4136	4141	4145	4157	
	4160	4247												
DAO = 001000	765#													
DAOTST = 015514	2937#													
DAOTT = 015672	2972#													
DB = 000210	746#	4269												
DCK = 100000	766#													
DCKTST = 014364	2753#													
DISPLA = 001030	686#	3710*	3714*	3945*	3952*	3959*	3966*	3973*	3977*	3983*	3990*	3997*	4053*	
	4346#													
DISPRE = 000174	607#	4346												
DKADR = 015054	2847#													
DLT = 100000	759#	1434												
DONCS = 012236	2363	2383#												
DONE = 021762	883	3622#												
DONEE = 002016	872	883#												
DONINT = 020760	3459	3463#												
DRCLR = 005754	1619#													
DRY = 000200	760#	2040	2050											
DS = 000040	742#	915	1527	1573	1593	1611	1680	1686	1701	1721	1724	1771	1823	
	1853	1944	1953	1989	2005	2042	2046	2054	2057	2076	2084	2116	2119	
	2144	2147	2184	2187	2223	2226	2259	2284	2287	2302	2309	2381	2420	
	2481	2558	2616	2625	2651	2654	2661	2682	2685	2707	2710	2738	2741	
	2745	2798	2801	2878	2907	2926	2956	2963	2991	2998	3014	3043	3076	
	3085	3112	3145											

H09

MAINDEC-11-DZRSB-E RH11-RS03LA-RS03-RS04 BASIC FUNCTION DIAGNOSTIC MACY11 27(732) 25-SEP-76 09:06 PAGE 113
 DZRSBE.P11 CROSS REFERENCE TABLE -- USER SYMBOLS

EXGEN	026774	4324	4329#											
EXREF	025664	4185	4187#											
EXTRP	020412	3300	3376	3382	3387#									
EXTTRP	020420	3295	3389#											
EXTTST	017676	3292#												
EXTT1	020220	3349#												
EXT1	020422	3388	3390#											
FILCHR	001014	680#	3658	3659										
FINDM	025632	1811	4182#											
FINTST	021066	3481	3486#											
FLOTBA	003052	1083#												
FLCTDA	003352	1158#												
FLOTWC	003212	1121#												
FUNDO	016620	3124#												
GOOD	=%000001	670#	1048*	1083*	1085	1087	1090*	1121*	1123	1125	1128*	1158*	1160	1162
		1165*	1258*	1272*	1287*	1314*	1315*	1317	1355*	1356*	1357	1368*	1369	1381*
		1382*	1384	1389*	1391	1394*	1396	1408*	1409*	1411	1414*	1415*	1416	1419
		1441*	1442	1450*	1451	1452*	1455*	1457	1458*	1464*	1466	1469*	1472*	1474
		1478*	1627*	1628*	1629	1687*	1688*	1690	1694*	1695*	1696	1728*	1729*	1731
		1735*	1736*	1737	1741*	1742	1763*	1764*	1766	1779*	1780*	1781	1824*	1825*
		1827	1831*	1832*	1833	1860*	1861*	1863	1867*	1868*	1869	1874*	1875	1895*
		1896*	1898	1911*	1912*	1913	1954*	1955*	1957	1961*	1962*	1963	1982*	1984
		2006*	2007*	2009	2017*	2018*	2019	2047*	2048*	2059*	2061*	2066	2070	2106*
		2109*	2111	2121*	2122*	2123	2126*	2127*	2129	2174*	2177*	2179	2194*	2195*
		2196	2207*	2208*	2209	2243*	2244	2247*	2248*	2250	2267*	2269	2272*	2273*
		2275	2327*	2328*	2333	2348*	2351*	2374*	2375*	2376	2413*	2414*	2415	2467*
		2468*	2469	2473*	2476*	2483*	2484*	2485	2489*	2490*	2492	2544*	2545*	2546
		2550*	2553*	2560*	2561*	2562	2566*	2567*	2569	2605*	2606*	2608	2618*	2620
		2803*	2804*	2805	2808*	2810	2813*	2816*	2820*	2822*	2824	2896*	2899*	2903
		2909*	2910	3007*	3008*	3009	3036*	3037*	3038	3068*	3069*	3070	3078*	3079*
		3080	3091*	3092*	3093	3138*	3139*	3140	3173*	3174*	3176	3280*	3281*	3283
		3345*	3383*	3436*	3450*	3544*	3545	3552*	3553*	3555	3559*	3560*	3561	4220
		4120#												
		4299	4301	4311	4325*	4331#								
		4299*	4333#											
		642#	911	915	919	924	930	936	942	948	954	973	977	982
		988	994	999	1004	1017	1021	1025	1033	1043	1049	1053	1057	1062
		1070	1074	1078	1089	1100	1108	1112	1116	1127	1137	1145	1149	1153
		1164	1174	1182	1186	1193	1199	1206	1216	1223	1231	1239	1259	1273
		1288	1303	1307	1319	1327	1331	1339	1343	1359	1362	1366	1371	1388
		1393	1398	1413	1421	1424	1436	1444	1468	1476	1492	1495	1498	1501
		1504	1507	1517	1524	1527	1530	1533	1536	1539	1542	1555	1558	1561
		1564	1567	1570	1573	1583	1590	1593	1596	1599	1602	1605	1608	1611
		1631	1634	1637	1642	1645	1648	1651	1665	1668	1672	1674	1677	1680
		1683	1686	1692	1698	1701	1716	1718	1721	1724	1727	1733	1739	1744
		1760	1762	1768	1771	1774	1777	1783	1804	1806	1814	1817	1820	1823
		1829	1835	1848	1850	1853	1856	1859	1865	1871	1877	1892	1894	1900
		1903	1906	1909	1915	1941	1944	1947	1950	1953	1959	1965	1981	1986
		1989	1993	2005	2011	2014	2021	2037	2042	2046	2054	2057	2069	2073
		2076	2079	2082	2084	2113	2116	2119	2125	2131	2134	2137	2140	2144
		2147	2181	2184	2187	2190	2198	2201	2204	2211	2216	2219	2223	2226
		2246	2252	2255	2259	2271	2277	2280	2284	2287	2290	2302	2306	2309
		2321	2338	2355	2378	2381	2417	2420	2471	2477	2481	2487	2494	2497
		2548	2554	2558	2564	2571	2574	2598	2601	2604	2610	2613	2616	2622
		2625	2642	2645	2648	2651	2654	2658	2661	2676	2679	2682	2685	2701
		2704	2707	2710	2732	2735	2738	2741	2745	2787	2795	2798	2801	2807

HALT 025150
 HINUM 027000
 HISAV 027004
 HLT = 104000

		2812	2826	2830	2853	2856	2875	2878	2881	2885	2907	2912	2915	2918
		2922	2926	2950	2953	2956	2959	2963	2966	2985	2988	2991	2994	2998
		3011	3014	3018	3022	3029	3034	3040	3043	3046	3049	3052	3055	3058
		3072	3076	3082	3085	3089	3095	3112	3115	3118	3136	3142	3145	3148
		3152	3171	3178	3181	3184	3188	3191	3210	3213	3216	3219	3223	3285
		3321	3325	3337	3341	3347	3360	3363	3375	3379	3385	3407	3413	3432
		3437	3457	3458	3462	3480	3484	3547	3548	3551	3557	3563	3568	3574
		3578	3597	3601	3604	4056	4087	4091	4096	4100	4105	4109	4114	4118
		4123	4127	4132	4136	4141	4145	4157	4160	4164	4174	4179		
HLTADR	001012	679#	3737*	3738*	3739	3740	3754							
IAERR	015130	2865#												
ICNT	001000	675#	805*	3618*	3688	3703	3705	3707*	3708*	3710	3713*	3714	3720	3749*
IE	= 000100	768#												
ILFDN	010762	2101	2149#											
ILFDNE	011332	2164	2228#											
ILFTB2	027342	2161	4407#											
ILF67	011334	2234#												
ILLFUN	010764	2160#												
ILLTAB	027300	2099	4389#											
ILLS1	010506	2096#												
INBUF	030164	4417#												
INCH	014576	2785#												
INPUT	023402	3855	3890	3891	3914#									
INTDON	020642	3433	3438#											
INTR4	020644	3444#												
INTR5	020540	3421#												
INT112	020634	3424	3436#											
INT114	020746	3446	3460#											
IR	= 000100	754#												
KBDIN	= 104416	3685	3729	3935#	3943	3950	3957	3964	3971	3981	3988	3995	4002	4057
KIPAR0	= 172340	716#	3302*											
KIPAR1	= 172342	717#	3303*											
KIPAR2	= 172344	718#	3304*	3350*										
KIPAR7	= 172356	719#	3301*											
KIPDR0	= 172300	720#	3305*											
KIPDR1	= 172302	721#	3306*											
KIPDR2	= 172304	722#	3307*											
KIPDR7	= 172316	723#	3308*											
LA	= 000204	745#	2321	2338	4279									
LAD	001010	678#	806*	1249*	1263*	1277*	3709*	3715	3717	3720				
LASTSC	017166	3197#												
LATDON	012046	2335	2339#											
LATST	011724	2319#												
LBT	= 002000	762#												
LONUM	026776	4298	4300	4309	4321*	4330#								
LOP1	003772	1249	1250#											
LOP2	004042	1263	1264#											
LOP3	004112	1277	1278#											
LOSAV	027002	4298*	4332#											
MAXREF	025716	4182	4193#											
MAXRF1	025720	4191	4194#											
MEMOUT	020426	3391#												
MODDON	021732	3615#												
MODNUM	021070	3496#												
MPRO	= 172100	780#	2392											
MR	= 000220	747#	942	999	1216	1223	1231	1239	1642	2037	2355	2381	4274	


```
PCNT      001004
PGE       = 002000
PGETST    013416
PGTRAP    020522
PIP       = 020000
PRTP      012426
PRTP1     012430
```

819#	814	818#	892	896#	955	959#	1005	1009#	1026	1027#	1035	1039#
641#	1065#	1080	1081#	1092	1093#	1101	1102#	1117	1118#	1131	1132#	1138
1064	1154	1155#	1168	1169#	1175	1176#	1188	1189#	1194	1195#	1200	1201#
1139#	1208#	1217	1218#	1224	1225#	1232	1233#	1240	1244#	1293	1297#	1309
1207	1321	1322#	1333	1334#	1345	1349#	1372	1376#	1399	1400#	1445	1446#
1310#	1484#	1508	1512#	1543	1547#	1574	1578#	1612	1616#	1653	1657#	1704
1480	1745	1749#	1791	1795#	1836	1840#	1878	1882#	1920	1924#	1970	1974#
1708#	1998#	2023	2027#	2086	2090#	2150	2154#	2229	2233#	2292	2296#	2314
1994	2322	2323#	2340	2344#	2384	2388#	2433	2437#	2506	2510#	2584	2588#
2318#	2630#	2663	2667#	2688	2692#	2712	2716#	2746	2750#	2842	2846#	2858
2626	2889	2893#	2932	2936#	2967	2971#	2999	3003#	3059	3063#	3096	3100#
2862#	3124#	3153	3157#	3192	3196#	3228	3232#	3287	3291#	3393	3397#	3416
3120	3439	3443#	3465	3469#	3487	3491#	4061	4065#				
3420#	2920	2924										
757#	2928#											
2901												
2894#												
2927	2929	2931#										
1513#												
1579#												
1485#												
1548#												
2421	2427	2431#										
865	882	888#										
1916#	4206											
1999#												
690#	808#	888#	1487	1502	1519	1534	1550	1565	1585	1600	1660	1666
1694	1711	1735	1755	1779	1797	1799	1831	1843	1867	1872	1885	1887
1911	1916	1917#	1931	1961	1976	1982	2030	2077	2102	2132	2165	2199
2212	2236	2243	2261	2267	2444	2460	2483	2518#	2521	2537	2560	2582#
2591	2632	2669	2694	2718	2764	2771	2788	2832	2849	2867	2947	2982
3024	3050	3104	3113	3127	3161	3207	3234	3249	3261	3267	3429	3452
3477	3526	3559	4005	4186#	4187	4190	4196#					
4362	4364	4366	4370	4380	4382#							
617#	635#	775#	807#	809	832#	857	864#	871	889#	2449	2474	2498
2501#	2505#	2515	2526	2551	2575	2578#	2583#	2928	2931#	4199	4203#	4284#
2043	2083	2085#										
755#	1422											
3619#												
808	888	1246	1248	1262	1276	1486#	1518#	1549#	1584#	1659#	1710#	1740
1754#	1917	1930#	2404#	2443#	2451#	2453#	2456#	2458#	2472	2503#	2530#	2582

M3INDEC-11-DZRSB-E RH11-RS03LA-RS03-RS04 BASIC FUNCTION DIAGNOSTIC MACY11 27(732) 25-SEP-76 09:06 PAGE 116
DZRSBE.P11 CROSS REFERENCE TABLE -- USER SYMBOLS

PSW = 177776
PTDONE 026570
QES 020442
QESDON 020536
QQ = 000001

RSCS1B 001140

643#	644	896*	960*	3402*	3426*	3427	3448*	3449	3471*	3539*	3611*	3617*
4003*												
644#												
4222	4258	4262	4280	4284*								
3398#												
3408	3414#											
814#	892#	955#	1005#	1026#	1035#	1064#	1080#	1092#	1101#	1117#	1131#	1138#
1154#	1168#	1175#	1188#	1194#	1200#	1207#	1217#	1224#	1232#	1240#	1293#	1309#
1321#	1333#	1345#	1372#	1399#	1445#	1480#	1508#	1543#	1574#	1612#	1653#	1704#
1745#	1791#	1836#	1878#	1920#	1970#	1994#	2023#	2086#	2150#	2229#	2292#	2314#
2322#	2340#	2384#	2433#	2506#	2584#	2626#	2663#	2688#	2712#	2746#	2842#	2858#
2889#	2932#	2967#	2999#	3059#	3096#	3120#	3153#	3192#	3228#	3287#	3393#	3416#
3439#	3465#	3487#	4061#									
1247	4298#											
4300#	4328											
4334#												
771#												
1245#												
3852	3933#											
3932#	4025	4028	4032	4377								
1709#												
1841#												
2631#												
2668#												
2693#												
2717#												
701#	826	952	1505	1515	1537	1568	1581	1603	1646	2066	2068	2070*
2071	2128	2288	2602	2726*	2916	3006	3015*	3016	3053	3077	3591	3607*
3989#	3990	4240										
697#	899#	922	963*	980	1067*	1068	1071*	1072	1075*	1076	1085*	1086
1096#	1097*	1098	1281	1286	1335*	1337	1340*	1341	1487*	1502	1519*	1534
1550*	1565	1585*	1600	1660*	1666	1693	1711*	1734	1755*	1778	1799*	1830
1843#	1866	1887*	1910	1931*	1960	1976*	1983	2002*	2030*	2077	2102*	2132
2165*	2199	2214	2238*	2242	2263*	2268	2405*	2444*	2460*	2482	2521*	2537*
2559	2591*	2632*	2669*	2694*	2718*	2764*	2771*	2788*	2823	2832*	2849*	2867*
2947*	2982*	3024*	3050	3104*	3113	3127*	3161*	3207*	3234*	3249*	3267*	3313*
3329*	3353*	3369*	3429*	3452*	3477*	3526*	3558	3951*	3952	4005*	4045*	4050*
4076*	4147*	4166*	4245									
711#	1336*											
694#	898*	917	962*	975	1014*	1015	1018*	1019	1022*	1023	1029*	1030*
1031	1299*	1301	1304*	1305	1360	1363*	1364	1489*	1490	1521*	1522	1552*
1553	1571	1587*	1588	1609	1624*	1643	1662*	1669*	1670	1678	1713*	1714
1722	1757*	1758	1769	1801*	1802	1815	1845*	1846	1854	1889*	1890	1901
1942	1979*											

RSCS2	001106	695*	820*	824*	835*	897*	904*	909	912*	961*	968*	970*	971	974*
		1041	1044*	1045	1047	1050*	1051	1054*	1055	1058*	1059*	1060	1063*	1311*
		1312*	1316	1354	1367	1383	1410	1422	1434	1625	1689	1730	1765	1809
		1826	1862	1897	1956	1978*	1991	2000*	2008	2016	2120	2193	2206	2249
		2274	2412	2466	2491	2543	2568	2607	2619	2765*	2783*	2784*	2802	2903*
		2908	2920	2924	3035	3065*	3067	3090	3102*	3106*	3125*	3131*	3137	3159*
		3166*	3172	3243*	3251*	3255*	3282	3533*	3535*	3541*	3554	3598*	3982*	3983
RSCS2B	001142	4038*	4209*	4210*	4233									
RSDA	001114	709*	1313*											
		698*	900*	928	964*	986	1142*	1143	1146*	1147	1150*	1151	1160*	1161
		1170*	1171*	1172	1267	1271	1499	1531	1562	1597	1621*	1632	1675	1719
		1772	1812	1851	1904	1945	2029*	2038*	2138	2188	2359*	2361*	2362	2374
		2462*	2539*	2640*	2646	2769*	2850*	2854	2868*	2904*	2923	2946*	2981*	3047
		3198*	3247*	3266*	3328*	3368*	3475*	3528*	3566	3572	3576	3958*	3959	4041*
		4077*	4084*	4088*	4097*	4106*	4115*	4124*	4133*	4142*	4148*	4167*	4250	
RSDB	001126	703*	966*	1353	1378*	1379*	1390	1395	1405*	1418	1430*	1440	1451*	1457*
		1465	1473	4272										
RSDS	001116	699*	913	1525	1591	1684	1821	1951	2040	2044	2050	2052	2074	2117
		2185	2300	2307	2649	2680	2705	2728	2736	2796	2876	2954	2961	2989
		2996	3041	3143	3179	3214	3221	3599	3965*	3966	4085	4089	4094	4098
		4103	4107	4112	4116	4121	4125	4130	4134	4139	4143	4155	4162	4177
		4260												
RSDT	001132	705*	836	840	844	848	850	4267						
RSER	001120	700*	821*	901*	934	965*	992	1179*	1180	1183*	1184	1190*	1191	1196*
		1197	1202*	1203*	1204	1493	1514*	1540	1556	1580*	1606	1622*	1635	1699
		1987	2114	2142	2182	2224	2253	2257	2278	2285	2299*	2304	2614	2643
		2656	2674*	2677	2702	2733	2793	2828	2873	2883	2905	2951	2964	2986
		3005*	3056	3074	3146	3182	3189	3211	3972*	3973	4158	4229		
RSLA	001124	702*	2319	2327	2331	2368	3977	4282						
RSMR	001130	704*	902*	940	969*	997	1211*	1212	1219*	1220*	1221	1226*	1227	1234*
		1235	1623*	1638	2033	2349	2353	2366	3066	3073*	3996*	3997	4277	
RSMRR	023712	631	3994*											
RSREG	026066	3743	4214*											
RSVCPS	001136	707*	3401*	3425*	3447*	3474*	3524*	3613	3614*					
RSVEC	001134	706*	3400*	3424*	3446*	3472*	3523*	3582*	3613*					
RSWC	001110	696*	903*	946	967*	1002	1105*	1106	1109*	1110	1113*	1114	1123*	1124
		1133*	1134*	1135	1253	1257	1323*	1325	1328*	1329	1488*	1496	1520*	1528
		1551*	1559	1586*	1594	1620*	1649	1661*	1663	1681	1712*	1725	1756*	1775
		1800*	1818	1844*	1857	1888*	1907	1932*	1948	1977*	2001*	2031*	2080	2103*
		2135	2166*	2202	2217	2235*	2260*	2406*	2445*	2461*	2522*	2538*	2592*	2611
		2633*	2670*	2695*	2719*	2762*	2772*	2789*	2809	2831*	2848*	2866*	2940*	2942*
		2945*	2975*	2977*	2980*	3025*	3044	3105*	3116	3128*	3162*	3201*	3203*	3206*
		3237*	3239*	3242*	3250*	3270*	3274*	3276*	3312*	3330*	3354*	3367*	3428*	3451*
		3476*	3527*	3549	3944*	3945	4004*	4042*	4078*	4149*	4168*	4255		
RSWCB	001144	710*	1324*											
RSWCWT	006626	1761*												
RS04DT	001162	776*	838*	842*	846*	854*	2754	2759	2814	2818	2938	2943	2973	2978
		3199	3204	3235	3240	3256	3268	3272	3564	3570				
RW	= 000006	724*	3305	3306	3307	3308								
RO	=%000000	646*	3628*	3812	3831*	4300*	4304*	4309*	4313*	4319*	4321	4322		
R1	=%000001	647*	819*	822*	1246*	1403*	1404*	1405	1406	1428*	1429*	1430	1431	1935*
		1936*	1938*	1939*	2395*	2396*	2397	3261*	3262*	3263*	3813	3823*	3824*	3830*
		3849	3853*	3866*	3868*	3870*	3873	3876*	4322*					
R2	=%000002	648*	2392*	2393	2394*	2398	2400*	2402*	2410*	2426	2428*	2773*	2774	2779
		3310*	3311*	3327*	3343	3346	3351*	3352*	3365*	3381	3384	3502*	3519*	3529*
		3535	3588*	3593*	3598	3814	3829*	3850	3856*	3857	3859	3861	3863*	3864*

M2INDEC-11-DZRSB-E RH11-RS03LA-RS03-RS04 BASIC FUNCTION DIAGNOSTIC MACY11 27(732) 25-SEP-76 09:06 PAGE 118
 DZRSBE.P11 CROSS REFERENCE TABLE -- USER SYMBOLS

R3	=%000003	3865*	3867*	3869*	3875*	4303*	4306*	4312*	4315*	4317*	4318*	2100	2161*	2163
		649#	1250*	1251*	1264*	1265*	1278*	1279*	1933*	1939	2099*	2099*	2161*	2163
		2634*	2635	2720*	2721	2774*	2775*	2776	2779*	2780*	2781	3500*	3520*	3521
R4	=%000004	3531	3537*	3815	3828*	3851	3855*	3874*	4302*	4307*				
		650#	1248*	1254	1255	1258	1260	1262*	1268	1269	1272	1274	1276*	1282
		1283*	1284	1287	1290	1926*	1927	1929*	1936	2635*	2636*	2637	2721*	2722*
		2723	3501*	3506	3508*	3519	3648	3658*	3662*	3676*	3767	3770*	3790	3796*
R5	=%000005	3816	3827*	4301*	4305*	4310*	4311*	4314*	4316*	4320*	4325	4326	1934*	1938
		651#	1247*	1253*	1254*	1255	1267*	1268*	1269	1281*	1282*	1284	1934*	1938
		3514*	3515*	3516*	3533	3649	3650*	3651	3653*	3654	3656	3664	3672*	3673*
		3674	3675*	3768	3769*	3776*	3778*	3780*	3795*	3817	3826*	3889	3890*	3891
		3911*	4329*											
SAVBAD	001100	689#												
SAVEE	001172	782#	3521*	3605*										
SC	= 100000	753#												
SCOPE	= 104400	817	895	958	1008	1026	1038	1064	1080	1092	1101	1117	1131	1138
		1154	1168	1175	1188	1194	1200	1207	1217	1224	1232	1243	1296	1309
		1321	1333	1348	1375	1399	1445	1483	1511	1546	1577	1615	1656	1707
		1748	1794	1839	1881	1923	1973	1997	2026	2089	2153	2232	2295	2317
		2322	2343	2387	2436	2509	2587	2629	2666	2691	2715	2749	2845	2861
		2892	2935	2970	3002	3062	3099	3123	3156	3195	3231	3290	3396	3419
		3442	3468	3490	3615	3622	3928#	4064	4204	4205				
SECT	017332	3232#												
SEEC	026422	4236	4263#											
SETPAT	016504	3101#												
SHIFT	026666	4304#	4308											
SILO	004632	1402#												
SILOB	004422	1352#												
SILOFL	005004	1448#												
SP	=%000006	652#	798*	3390*	3399*	3421*	3444*	3470*	3583*	3609*	3612*	3818	3822*	3942*
		3949*	3956*	3963*	3970*	3980*	3987*	3994*	4001*	4022*	4294*	4338*	4339*	4344
		4352	4353	4378										
SRAS	023666	630	3987#											
SRC52	023642	629	3980#											
SRSBA	023512	624	3949#											
SRSCS1	023736	632	4001#											
SRSDA	023536	625	3956#											
SRSDS	023562	626												

M2INDEC-11-DZRSB-E RH11-RS03LA-RS03-RS04 BASIC FUNCTION DIAGNOSTIC MACY11 27(732) 25-SEP-76 09:06 PAGE 119
 DZRSBE.P11 CROSS REFERENCE TABLE -- USER SYMBOLS

[illegible]

[illegible]

C10

 MAINDEC-11-DZRSB-E RH11-RS03LA-RS03-RS04 BASIC FUNCTION DIAGNOSTIC MACY11 27(732) 25-SEP-76 09:06 PAGE 121
 DZRSBE.P11 CROSS REFERENCE TABLE -- USER SYMBOLS

TYPES = 104406	862	878	3931#										
UNCMP 001156	774#	827*	833	852	866*	873*	1505	1515	1537	1568	1581	1603	1646
	2126	3007	3015	3053	3078								
UNITST 007544	1925#												
UNITSV 001154	773#	826*	833	852*	2897	3504							
UNNUM 001152	772#	828*	835	861	868*	874*	877	897	912	951	974	1063	1311
	1314	1356	1382	1409	1627	1687	1728	1763	1824	1860	1895	1927	1934
	1954	2006	2017	2058	2121	2194	2207	2247	2272	2414	2467	2489	2544
	2566	2605	2784	2803	3037	3069	3091	3139	3174	3280	3506	3514	3541
	3552	4026*	4073*	4210									
UP = 000000	725#	3305	3306	3307	3308								
WAITRY 026600	1673	1717	1761	1805	1849	1893	1940	1980	2004	2597	2641	2675	2700
	2852	2949	2984	3209	3431	3479	4287#						
WC = 000010	740#	948	1004	1108	1112	1116	1137	1327	1331	1498	1530	1561	1596
	1651	1665	1683	1727	1777	1820	1859	1909	1950	2082	2137	2204	2219
	2613	3046	3118	3551	4056	4252							
WCE = 040000	758#												
WCEDNB 013356	2536	2578#											
WCEDNE 012772	2459	2501#											
WCEDON 013014	2454	2505#											
WCEDOS 013400	2517	2531	2582#										
WCETB 013054	2521#	2581											
WCETSB 013024	2515#												
WCETST 012462	2442#												
WCETT 012470	2444#	2504											
WORK 001174	794#	1212*	1213*	1214	1227*	1228*	1229	1235*	1236*	1237	1245*	1380*	1386*
	1638#	1639*	1640	1797*	1798*	1842*	1872*	1873	1885*	1886*	2032*	2035*	2058*
	2062	2064*	2162*	2171*	2191	2212*	2213*	2214	2236*	2237*	2238	2261*	2262*
	2263	2326*	2329*	2360*	2364*	2395	2405	2411*	2727*	2730*	2895*	2897	2900*
	2923*	3258*	3260*	3264*	3404*	3405*	3454*	3455*	3503*	3504	3510*	3590*	3591
	3595*	3607	4029*	4030*	4042	4287*	4290*	4291	4323*	4327*			
WORK1 001176	795#	2167*	2168*	2169	2370*	2372*	3499*	3512	3518*	3580	4039*	4040*	4041
WORK2 001200	796#	2756*	2758*	2761*	2762	2789	4036*	4058					
WRCKT 006564	1753#												
WRCKTB 007350	1884#												
WRTLCK 024354	614	4066#											
WRTST 006150	1658#												
WRTSTB 006736	1796#	4192											
WT 024272	4047#	4052											
WTDV 021336	3540#												
WTDV1 021706	3581	3606	3611#										
WTDV2 021566	3583	3586#	3609										
WTE 024302	4044	4050#											
WMP = 000004	779#	2394											
XXX 027114	4337	4354#											
ZERONE 004524	1377#												
SENDAD 022026	602	3631#											
SEND1 022036	3629	3633#											
. = 030764	600#	601#	603#	606#	610#	613#	616#	622#	634#	673#	688#	810	831
	859	880	881#	910	914	918	923	929	935	941	947	953	972
	976	981	987	993	998	1003	1016	1020	1024	1032	1042	1052	1056
	1061	1069	1073	1077	1088	1099	1107	1111	1115	1126	1136	1144	1148
	1152	1163	1173	1181	1185	1192	1198	1205	1215	1222	1230	1238	1302
	1306	1318	1326	1330	1338	1342	1358	1361	1365	1370	1392	1397	1412
	1423	1432	1435	1443	1453	1467	1470	1475	1491	1494	1497	1500	1503
	1506	1516	1523	1526	1529	1532	1535	1538	1541	1554	1557	1560	1563

[illegible]

E10

MAINDEC-11-DZRSB-E RH11-RS03LA-RS03-RS04 BASIC FUNCTION DIAGNOSTIC MACY11 27(732) 25-SEP-76 09:06 PAGE 123
DZRSBE.P11 CROSS REFERENCE TABLE -- USER SYMBOLS

.TYPEB	022556	3760#	
.TYPE0	022566	3762#	3930
.TYPES	022576	3764#	3931

MAINDEC-11-DZRSB-E RH11-RS03LA-RS03-RS04 BASIC FUNCTION DIAGNOSTIC MACY11 27(732) 25-SEP-76 09:06 PAGE 125
DZRSBE.P11 CROSS REFERENCE TABLE -- MACRO NAMES

END	1#	584#													
LDPDR	727#	3305	3306	3307	3308										
NEWST	1#	584#	814	892	955	1005	1026	1035	1064	1080	1092	1101	1117	1131	1138
	1154	1168	1175	1188	1194	1200	1207	1217	1224	1232	1240	1293	1309	1321	1333
	1345	1372	1399	1445	1480	1508	1543	1574	1612	1653	1704	1745	1791	1836	1878
	1920	1970	1994	2023	2086	2150	2229	2292	2314	2322	2340	2394	2433	2506	2584
	2626	2663	2688	2712	2746	2842	2858	2889	2932	2967	2999	3059	3096	3120	3153
	3192	3228	3287	3393	3416	3439	3465	3487	4061						
ODT11	1#														
ODT11X	1#														
POP	1#	670#	3696	3699	3795	3826	3874								
PRINT	1#	670#	831	859	880	2337	2930	3627	3736	3742	3834	3879	3899	3903	4024
	4027	4031	4067	4082	4092	4101	4110	4119	4128	4137	4175	4201	4216	4219	4224
	4228	4232	4239	4244	4249	4254	4259	4266	4271	4276	4281	4371	4373	4376	
PUSH	1#	670#	3693	3766	3812	3849									
SCOPE.	1#	814#	892#	955#	1005#	1026#	1035#	1064#	1080#	1092#	1101#	1117#	1131#	1138#	1154#
	1168#	1175#	1188#	1194#	1200#	1207#	1217#	1224#	1232#	1240#	1293#	1309#	1321#	1333#	1345#
	1372#	1399#	1445#	1480#	1508#	1543#	1574#	1612#	1653#	1704#	1745#	1791#	1836#	1878#	1920#
	1970#	1994#	2023#	2086#	2150#	2229#	2292#	2314#	2322#	2340#	2384#	2433#	2506#	2584#	2626#
	2663#	2688#	2712#	2746#	2842#	2858#	2889#	2932#	2967#	2999#	3059#	3096#	3120#	3153#	3192#
	3228#	3287#	3393#	3416#	3439#	3465#	3487#	4061#							
SET	3915#	3928	3929	3930	3931	3932	3933	3934	3935	3936	3937				
TYPADR	1#														
TYPBIT	1#	670#													
TYPCHR	1#	670#													
TYPDEC	1#	670#													
TYPLIN	1#	670#													
TYPOCS	1#	670#	860	877											
TYPOCT	1#	670#	3740	4217	4220	4225	4229	4233	4240	4245	4250	4255	4260	4267	4272
	4277	4282	4374												
TYPTXT	1#	670#	831	859	880	2336	2930	4024	4027	4031	4066	4082	4092	4101	4110
	4119	4128	4137	4175	4201	4216	4219	4223	4227	4231	4239	4244	4249	4254	4259
	4266	4271	4276	4281	4371	4372	4376								
SCATCH	1#	584#	600												
SCMTAG	1#	584#	673												
SDONE	1#	584#	3620												
SEQUAT	1#	584#	641												
SHLT	1#	584#	3721												
SKBDIN	1#	584#	4360												
SKMMR	1#														
SOCTAL	1#	584#	3754												
SPOWER	1#	584#	3802												
SRAND2	1#														
SRAND4	1#														
SRDDEC	1#														
SRDLIN	1#	584#	3882												
SRDOCT	1#	584#	3842												
SSCOPE	1#	584#	3678												
\$SET	3915#	3928	3929	3930	3931	3932	3933	3934	3935	3936	3937				
\$SETUP	1#	584#	798												
\$SMMR	1#														
\$SUSMR	1#	584#	4335												
\$SMDOC	1#	584#	588												
\$TRAP	1#	584#	3915												
\$TYPE	1#	584#	3638												
\$TYPEA	1#														

G10

MAINDEC-11-DZRSB-E RH11-RS03LA-RS03-RS04 BASIC FUNCTION DIAGNOSTIC MACY11 27(732) 25-SEP-76 09:06 PAGE 126
DZRSBE.P11 CROSS REFERENCE TABLE -- MACRO NAMES

STYPED 18
SUMMR 18

MAINDEC-11-DZRSB-E RH11-RS03LA-RS03-RS04 BASIC FUNCTION DIAGNOSTIC MACY11 27(732) 25-SEP-76 09:06 PAGE 128
DZRSBE.P11 CROSS REFERENCE TABLE -- PERMANENT SYMBOLS

ADC	3624	4310	4312	4314	4315	4317	4320								
ADD	1695	1736	1780	1832	1868	1912	1962	2213	2237	2262	2428	2484	2561	2816	2820
	2822	3560	3623	3669	3672	3925	4186	4294	4309	4311	4313	4316	4318	4319	
ASL	866	4304													
BCC	1091	1129	1166	2454	2531	3511	3596								
BCS	867	1453	1459	1470	1479	2459	2536	2901							
BEQ	810	823	834	849	851	858	910	914	918	923	929	935	941	947	953
	972	976	981	987	993	998	1003	1016	1020	1024	1032	1042	1046	1052	1056
	1061	1069	1073	1077	1088	1099	1107	1111	1115	1126	1136	1144	1148	1152	1163
	1173	1181	1185	1192	1198	1205	1215	1222	1230	1238	1252	1256	1266	1270	1280
	1285	1302	1306	1318	1326	1330	1338	1342	1358	1361	1365	1370	1385	1392	1397
	1412	1417	1420	1423	1432	1443	1467	1475	1491	1494	1497	1500	1503	1506	1523
	1526	1529	1532	1535	1541	1554	1557	1560	1563	1566	1569	1572	1589	1592	1595
	1598	1601	1607	1610	1630	1633	1636	1641	1644	1647	1650	1664	1667	1676	1679
	1682	1685	1691	1697	1700	1720	1723	1726	1732	1738	1743	1767	1770	1773	1776
	1782	1810	1813	1816	1819	1822	1828	1834	1852	1855	1858	1864	1870	1876	1899
	1902	1905	1908	1914	1943	1946	1949	1952	1958	1964	1985	1988	1992	2010	2013
	2020	2041	2049	2051	2053	2056	2063	2067	2072	2075	2078	2081	2101	2112	2115
	2118	2124	2130	2133	2136	2139	2143	2146	2164	2180	2183	2186	2189	2197	2200
	2203	2210	2215	2218	2222	2225	2245	2251	2254	2258	2270	2276	2279	2283	2286
	2289	2301	2305	2308	2320	2330	2334	2350	2354	2363	2365	2367	2377	2416	2419
	2427	2450	2470	2475	2480	2486	2493	2496	2499	2516	2527	2547	2552	2557	2563
	2570	2573	2576	2600	2603	2609	2615	2621	2624	2644	2650	2653	2657	2660	2678
	2681	2684	2703	2706	2709	2734	2737	2740	2744	2794	2797	2800	2806	2811	2825
	2829	2855	2874	2877	2880	2884	2898	2906	2911	2914	2917	2921	2952	2955	2962
	2965	2987	2990	2997	3010	3013	3017	3021	3028	3033	3039	3042	3045	3048	3051
	3057	3071	3075	3081	3084	3088	3094	3111	3114	3117	3135	3141	3144	3147	3151
	3170	3177	3180	3183	3187	3190	3205	3212	3215	3222	3257	3273	3279	3284	3324
	3340	3344	3362	3378	3382	3412	3461	3483	3505	3546	3550	3556	3562	3567	3573
	3577	3581	3600	3603	3606	3629	3655	3687	3689	3704	3716	3731	3783	3786	3858

	3265	3269	3406	3456	3507	3513	3532	3538	3565	3571	3592	3626	3652	3657	3663
	3671	3691	3702	3735	3748	3775	3789	3791	3825	3898	3902	3909	4059	4185	4198
	4200	4215	4236	4308	4328	4337	4348	4350	4362	4364	4370				
BPL	1671	1715	1759	1803	1847	1891	2108	2176	2241	2266	2409	2448	2465	2525	2542
	2768	2792	2835	2993	3031	3109	3133	3168	3218	3246	3254	3318	3320	3334	3336
	3357	3359	3372	3374	3661	3666	3745	3894	4010	4048	4055	4081	4152	4154	4171
	4366														
BR	825	839	843	847	853	865	869	882	1261	1275	1289	1291	1433	1454	1471
	2043	2065	2083	2335	2379	2401	2455	2500	2532	2577	2757	2817	2821	2872	2902
	2927	2941	2976	3202	3238	3259	3271	3275	3322	3326	3338	3342	3376	3388	3433
	3459	3481	3517	3522	3534	3569	3575	3667	3692	3697	3700	3761	3763	3772	3838
	3872	3881	3900	3905	3946	3953	3960	3967	3974	3978	3984	3991	3998	4011	4049
	4052	4060	4071	4293	4343										
CLC	1084	1122	1159	1449	1463	2452	2529	3509	3589	3594					
CLR	805	806	820	842	846	874	1030	1059	1097	1134	1171	1203	1220	1378	1389
	1403	1414	1428	1710	1842	1926	2032	2038	2059	2162	2400	2410	2411	2432	2443
	2456	2462	2473	2520	2533	2539	2550	2727	2769	2770	2896	2904	3247	3263	3266
	3302	3327	3328	3365	3368	3387	3392	3454	3499	3500	3501	3525	3588	3614	3764
	3823	3853	3854	4075	4165	4189	4195	4287	4303						
CLRB	1313	3749	3771	3773	3793										
CMP	836	840	844	848	850	909	913	917	946	971	1002	1015	1019	1023	1031
	1041	1045	1051	1055	1060	1068	1072	1076	1087	1106	1110	1114	1125	1143	1147
	1151	1162	1180	1184	1191	1197	1214	1229	1237	1255	1269	1284	1301	1305	1317
	1325	1329	1337	1341	1357	1360	1369	1384	1391	1396	1406	1411	1416	1419	1431
	1442	1466	1474	1496	1502	1525	1528	1534	1540	1559	1565	1571	1591	1594	1600
	1606	1609	1629	1640	1643	1649	1663	1666	1675	1678	1684	1690	1696	1719	1722
	1731	1737	1742	1766	1769	1772	1781	1812	1815	1821	1827	1833	1851	1854	1863
	1869	1875	1898	1901	1904	1913	1927	1942	1945	1951	1957	1963	1984	2009	2012
	2019	2052	2055	2066	2074	2077	2080	2111	2114	2117	2123	2129	2132	2135	2145
	2169	2179	2182	2185	2196	2199	2202	2209	2214	2217	2221	2244	2250	2253	2269
	2275	2278	2282	2300	2307	2333	2362	2376	2415	2418	2425	2426	2469	2479	2485
	2492	2495	2546	2556	2562	2569	2572	2599	2608	2620	2623	2637	2643	2646	2649
	2652	2659	2677	2680	2683	2702	2705	2708	2723	2736	2739	2743	2754	2776	2781
	2793	2796	2799	2805	2810	2814	2824	2854	2873	2876	2879	2910	2913	2938	2951
	2954	2961	2973	2989	2996	3009	3012	3020	3032	3038	3041	3044	3050	3056	3070
	3074	3080	3083	3087	3093	3110	3113	3116	3134	3140	3143	3150	3159	3176	3179
	3182	3186	3199	3214	3221	3235	3268	3283	3323	3339	3343	3361	3377	3381	3410
	3411	3460	3482	3485	3506	3545	3555	3561	3564	3566	3572	3576	3599	3602	3670
	3698	3790	3891	4058	4068	4085	4089	4094	4098	4103	4107	4112	4116	4121	4125
	4130	4134	4139	4143	4155	4158	4162	4177	4190	4193	4341	4344	4347	4363	
CMPB	3656	3688	3705	3857	3859	3861	3897	3901	3908	4369					
COM	4030														
DEC	822	1251	1265	1279	1386	2035	2048	2064	2329	2351	2364	2372	3264	3405	3537
	3605	3662	4307	4323	4327										
EMT	642														
HALT	601	881	3746	3820	3837	4070	4083	4093	4102	4111	4120	4129	4138	4176	
INC	824	868	1404	1415	1429	1929	2361	2730	2899	3455	3508	3515	3520	3593	3733
	3824	3871	4290												
INCB	3708	3713	3784	3788											
JMP	611	614	618	623	624	625	626	627	628	629	630	631	632	636	811
	883	1811	2148	2205	2227	2421	2429	2504	2517	2581	3408	3619	3633	3751	4192
	4206	4222	4262												
JSR	1247	1673	1717	1761	1805	1849	1893	1940	1980	2004	2597	2641	2675	2700	2852
	2949	2984	3209	3431	3479	3631	3743								
MOV	798	799	800	801	802	803	804	808	819	821	826	827	828	835	838
	861	873	877	888	890	891	896	897	898	899	900	901	902	903	904

	912	960	961	962	963	964	965	966	967	968	969	970	974	1013	1014
	1018	1022	1029	1044	1047	1048	1050	1054	1058	1063	1067	1071	1075	1083	1085
	1086	1096	1105	1109	1113	1121	1123	1124	1133	1142	1146	1150	1158	1160	1161
	1170	1179	1190	1196	1202	1211	1212	1219	1226	1227	1234	1235	1245	1246	1248
	1249	1250	1253	1254	1257	1258	1262	1263	1264	1267	1268	1271	1272	1276	1277
	1278	1281	1282	1286	1287	1299	1311	1314	1316	1323	1335	1353	1354	1355	1363
	1367	1379	1380	1381	1383	1390	1394	1395	1405	1408	1410	1418	1430	1440	1441
	1450	1451	1455	1457	1464	1465	1472	1473	1486	1487	1488	1489	1514	1518	1519
	1520	1521	1549	1550	1551	1552	1580	1584	1585	1586	1587	1620	1621	1622	1623
	1624	1625	1627	1638	1659	1660	1661	1662	1687	1689	1693	1694	1711	1712	1713
	1728	1730	1734	1735	1740	1741	1754	1755	1756	1757	1763	1765	1778	1779	1797
	1798	1799	1800	1801	1824	1826	1830	1831	1843	1844	1845	1860	1862	1866	1867
	1872	1873	1874	1885	1886	1887	1888	1889	1895	1897	1910	1911	1916	1917	1930
	1931	1932	1933	1934	1935	1936	1938	1939	1954	1956	1960	1961	1976	1977	1979
	1982	1983	2001	2002	2003	2006	2008	2015	2016	2017	2029	2030	2031	2039	2047
	2058	2068	2070	2096	2097	2099	2100	2102	2103	2104	2106	2110	2120	2121	2126
	2128	2161	2163	2165	2166	2167	2171	2172	2174	2178	2193	2194	2206	2207	2212
	2235	2236	2238	2239	2242	2243	2247	2249	2260	2261	2263	2264	2267	2268	2272
	2274	2281	2297	2299	2326	2327	2331	2345	2346	2348	2359	2360	2368	2370	2374
	2389	2390	2391	2392	2394	2395	2396	2402	2404	2405	2406	2407	2412	2413	2431
	2444	2445	2446	2451	2460	2461	2463	2466	2467	2472	2476	2482	2483	2489	2491
	2503	2518	2521	2522	2523	2528	2537	2538	2540	2543	2544	2544	2553	2559	2560
	2566	2568	2580	2582	2590	2591	2592	2593	2596	2605	2607	2611	2632	2633	2634
	2635	2639	2640	2669	2670	2671	2674	2694	2695	2696	2699	2718	2719	2720	2721
	2725	2726	2756	2758	2761	2762	2763	2764	2766	2771	2772	2773	2774	2778	2779
	2783	2784	2788	2789	2790	2802	2803	2808	2809	2813	2823	2831	2832	2833	2848
	2849	2850	2851	2866	2867	2868	2869	2895	2903	2908	2923	2940	2942	2945	2946
	2947	2948	2975	2977	2980	2981	2982	2983	3005	3006	3007	3015	3023	3024	3025
	3026	3035	3036	3067	3068	3073	3077	3078	3090	3091	3103	3104	3105	3107	3126
	3127	3128	3129	3137	3138	3160	3161	3162	3163	3172	3173	3198	3201	3203	3206
	3207	3208	3233	3234	3237	3239	3242	3244	3248	3249	3250	3252	3258	3260	3261
	3262	3267	3270	3274	3276	3277	3280	3282	3293	3294	3295	3296	3300	3301	3303
	3304	3305	3306	3307	3308	3309	3310	3311	3312	3313	3314	3329	3330	3331	3345
	3346	3350	3351	3352	3353	3354	3355	3367	3369	3370	3383	3384	3390	3391	3399
	3400	3401	3402	3403	3404	3421	3422	3424	3425	3426	3427	3428	3429	3430	3436
	3444	3446	3447	3448	3449	3450	3451	3452	3453	3470	3471	3472	3474	3475	3476
	3477	3478	3497	3498	3502	3503	3514	3519	3521	3523	3524	3526	3527	3528	3529
	3530	3533	3535	3536	3539	3541	3542	3544	3552	3554	3558	3559	3579	3582	3583
	3584	3590	3598	3607	3608	3609	3611	3612	3613	3616	3617	3618	3628	3648	3649
	3650	3653	3668	3674	3675	3676	3693	3694	3696	3699	3709	3710	3714	3717	3737
	3740	3760	3767	3768	3769	3770	3795	3796	3797	3798	3810	3811	3812	3813	3814
	3815	3816	3817	3818	3819	3822	3826	3827	3828	3829	3830	3831	3832	3833	3847
	3848	3849	3850	3851	3855	3873	3874	3875	3876	3889	3890	3911	3922	3924	3926
	3942	3944	3945	3949	3951	3952	3956	3958	3959	3963	3965	3966	3970	3972	3973
	3977	3980	3982	3983	3987	3989	3990	3994	3996	3997	4001	4003	4004	4005	4006
	4022	4026	4029	4033	4036	4039	4041	4042	4045	4046	4050	4051	4053	4073	4076
	4077	4078	4079	4084	4088	4097	4106	4115	4124	4133	4142	4146	4147	4148	4149
	4150	4166	4167	4168	4169	4182	4183	4188	4194	4196	4209	4210	4217	4220	4225
	4229	4233	4240	4245	4250	4255	4260	4267	4272	4277	4282	4298	4299	4300	4301
	4302	4321	4322	4325	4326	4338	4339	4340	4345	4346	4352	4353	4367	4374	4378
	4381														
MOV8	1183	1300	1304	1324	1328	1336	1340	2919	3658	3659	3664	3707	3739	3762	3765
	3792	3856	3895	3906											
NOP	1292	1937	3348	3389	4023										
RESET	905	1040	3630												
ROL	1090	1128	1165	1452	1458	1469	1478	2061	2453	2458	2530	2535	2900	3510	3595

K10

 MAINDEC-11-DZRSB-E RH11-RS03LA-RS03-RS04 BASIC FUNCTION DIAGNOSTIC MACY11 27(732) 25-SEP-76 09:06 PAGE 131
 DZRSBE.P11 CROSS REFERENCE TABLE -- PERMANENT SYMBOLS

	3776	3778	3780	3866	3867	3868	3869	3870	4305	4306					
ROLB	3777	3779	3781												
ROR	2371	3863	3864	3865											
RTI	3585	3610	3677	3711	3718	3750	3799	3835	3877	3912	4211	4355	4382		
RTS	4285	4295	4329												
SEC	1456	1477	2060	2457	2534										
SUB	3738	3923													
TRAP	3925	3928	3929	3930	3931	3932	3933	3934	3935	3936	3937				
TST	922	928	934	952	980	986	992	1098	1135	1172	1204	1260	1274	1290	1493
	1499	1531	1556	1562	1597	1632	1635	1681	1699	1725	1775	1818	1857	1907	1948
	1987	2062	2071	2138	2142	2188	2191	2224	2257	2285	2288	2304	2393	2397	2398
	2602	2611	2614	2656	2733	2759	2818	2828	2883	2905	2916	2943	2957	2964	2978
	2986	2992	3016	3047	3066	3146	3189	3204	3211	3217	3240	3256	3272	3297	3319
	3335	3358	3373	3531	3549	3570	3695	3715	3744	4054	4172	4187	4214	4291	4349
	4361	4379													
TSTB	1670	1714	1758	1802	1846	1890	2107	2175	2240	2265	2408	2447	2464	2524	2541
	2594	2672	2697	2767	2785	2791	2834	2870	3030	3108	3132	3164	3167	3245	3253
	3317	3333	3356	3371	3654	3660	3665	3703	3782	3785	3893	4009	4047	4080	4151
	4153	4170	4288	4365											
WAIT	3540	3586													
.ASCIZ	832	860	881	2338	2931	3628	3737	3743	3835	3881	3900	3905	4025	4028	4032
	4068	4083	4093	4102	4111	4120	4129	4138	4176	4202	4217	4220	4225	4229	4233
	4240	4245	4250	4255	4260	4267	4272	4277	4282	4372	4374	4377			
.BLKB	3914														
.BLKW	3637	3801	4416	4417											
.ENABL	1	584													
.END	4418														
.ENDC	670	799	805	817	895	958	1008	1026	1038	1064	1080	1092	1101	1117	1131
	1138	1154	1168	1175	1188	1194	1200	1207	1217	1224	1232	1243	1296	1309	1321
	1333	1348	1375	1399	1445	1483	1511	1546	1577	1615	1656	1707	1748	1794	1839
	1881	1923	1973	1997	2026	2089	2153	2232	2295	2317	2322	2343	2387	2436	2509
	2587	2629	2666	2691	2715	2749	2845	2861	2892	2935	2970	3002	3062	3099	3123
	3156	3195	3231	3290	3396	3419	3442	3468	3490	3625	3627	3628	3634	3691	3719
	3734	3736	3744	3752	3818	3826	3835	3836	3892	3915	3929	3930	3931	3932	3933
	3934	3935	3936	3937	3938	4064									
.EVEN	832	860	881	2338	2931	3628	3737	3743	3835	3881	3900	3905	4025	4028	4032
	4068	4083	4093	4102	4111	4120	4129	4138	4176	4202	4217	4220	4225	4229	4233
	4240	4245	4250	4255	4260	4267	4272	4277	4282	4372	4374	4377			
.IF	654	798	805	814	892	955	1005	1026	1035	1064	1080	1092	1101	1117	1131
	1138	1154	1168	1175	1188	1194	1200	1207	1217	1224	1232	1240	1293	1309	1321
	1333	1345	1372	1399	1445	1480	1508	1543	1574	1612	1653	1704	1745	1791	1836
	1878	1920	1970	1994	2023	2086	2150	2229	2292	2314	2322	2340	2384	2433	2506
	2584	2626	2663	2688	2712	2746	2842	2858	2889	2932	2967	2999	3059	3096	3120
	3153	3192	3228	3287	3393	3416	3439	3465	3487	3625	3626	3628	3633	3685	3719
	3729	3734	3743	3747	3818	3826	3834	3835	3891	3914	3928	3929	3930	3931	3932
	3933	3934	3935	3936	3937	4061									
.IFF	799	3626	3634	3691	3729	3747	3836	3892	3915	3928	3929	3930	3931	3932	3933
	3934	3935	3936	3937											
.IIF	591	592	593	594	595	596	597	598	599	600	3635	3711	3715	3720	3753
	3754	3914													
.IRP	3693	3696	3699	3767	3795	3812	3826	3849	3874	3928					
.LIST	1	584	601	670	814	818	832	860	881	892	896	955	959	1005	1009
	1026	1027	1035	1039	1064	1065	1080	1081	1092	1093	1101	1102	1117	1118	1131
	1132	1138	1139	1154	1155	1168	1169	1175	1176	1188	1189	1194	1195	1200	1201
	1207	1208	1217	1218	1224	1225	1232	1233	1240	1244	1293	1297	1309	1310	1321
	1322	1333	1334	1345	1349	1372	1376	1399	1400	1445	1446	1480	1484	1508	1512

	1543	1547	1574	1578	1612	1616	1653	1657	1704	1708	1745	1749	1791	1795	1836
	1840	1878	1882	1920	1924	1970	1974	1994	1998	2023	2027	2086	2090	2150	2154
	2229	2233	2292	2296	2314	2318	2322	2323	2338	2340	2344	2384	2388	2433	2437
	2506	2510	2584	2588	2626	2630	2663	2667	2688	2692	2712	2716	2746	2750	2842
	2846	2858	2862	2889	2893	2931	2932	2936	2967	2971	2999	3003	3059	3063	3096
	3100	3120	3124	3153	3157	3192	3196	3228	3232	3287	3291	3393	3397	3416	3420
	3439	3443	3465	3469	3487	3491	3628	3737	3743	3835	3881	3900	3905	3915	3928
	3929	3930	3931	3932	3933	3934	3935	3936	3937	3938	4025	4028	4032	4061	4065
	4068	4083	4093	4102	4111	4120	4129	4138	4176	4202	4217	4220	4225	4229	4233
	4240	4245	4250	4255	4260	4267	4272	4277	4282	4372	4374	4377			
.MACR	727														
.MACRO	1	3915													
.MCALL	584	670	814	892	955	1005	1026	1035	1064	1080	1092	1101	1117	1131	1138
	1154	1168	1175	1188	1194	1200	1207	1217	1224	1232	1240	1293	1309	1321	1333
	1345	1372	1399	1445	1480	1508	1543	1574	1612	1653	1704	1745	1791	1836	1878
	1920	1970	1994	2023	2086	2150	2229	2292	2314	2322	2340	2384	2433	2506	2584
	2626	2663	2688	2712	2746	2842	2858	2889	2932	2967	2999	3059	3096	3120	3153
.NLIST	3192	3228	3287	3393	3416	3439	3465	3487	4061						
	1	584	601	670	814	818	832	860	881	892	896	955	959	1005	1009
	1026	1027	1035	1039	1064	1065	1080	1081	1092	1093	1101	1102	1117	1118	1131
	1132	1138	1139	1154	1155	1168	1169	1175	1176	1188	1189	1194	1195	1200	1201
	1207	1208	1217	1218	1224	1225	1232	1233	1240	1244	1293	1297	1309	1310	1321
	1322	1333	1334	1345	1349	1372	1376	1399	1400	1445	1446	1480	1484	1508	1512
	1543	1547	1574	1578	1612	1616	1653	1657	1704	1708	1745	1749	1791	1795	1836
	1840	1878	1882	1920	1924	1970	1974	1994	1998	2023	2027	2086	2090	2150	2154
	2229	2233	2292	2296	2314	2318	2322	2323	2338	2340	2344	2384	2388	2433	2437
	2506	2510	2584	2588	2626	2630	2663	2667	2688	2692	2712	2716	2746	2750	2842
	2846	2858	2862	2889	2893	2931	2932	2936	2967	2971	2999	3003	3059	3063	3096
	3100	3120	3124	3153	3157	3192	3196	3228	3232	3287	3291	3393	3397	3416	3420
	3439	3443	3465	3469	3487	3491	3628	3737	3743	3835	3881	3900	3905	3915	3928
	3929	3930	3931	3932	3933	3934	3935	3936	3937	3938	4025	4028	4032	4061	4065
	4068	4083	4093	4102	4111	4120	4129	4138	4176	4202	4217	4220	4225	4229	4233
	4240	4245	4250	4255	4260	4267	4272	4277	4282	4372	4374	4377			
.PAGE	640	673	3678	3721	3754	3802	3842	3882							
.REM	1														
.REPT	601														
.SBTTL	814	892	955	1005	1035	1240	1293	1345	1372	1480	1508	1543	1574	1612	1653
	1704	1745	1791	1836	1878	1920	1970	1994	2023	2086	2150	2229	2292	2314	2340
	2384	2433	2506	2584	2626	2663	2688	2712	2746	2842	2858	2889	2932	2967	2999
	3059	3096	3120	3153	3192	3228	3287	3393	3416	3439	3465	3487	3620	3638	3678
.TITLE	3721	3754	3802	3842	3882	3915	4061								
	584														

ERRORS DETECTED: 0
DEFAULT GLOBALS GENERATED: 0

*,DZRSBE.SEG/SOL/CRF/PAGNUM/NL:TOC=DZRSBE.SML,DZRSBE.P11
RUN-TIME: 28 45 8 SECONDS
RUN-TIME RATIO: 204/82=2.4
CORE USED: 23K (45 PAGES)

Spooler runtime 18 Seconds, 77 KCS, 493 disk reads, 3 disk writes, 128 pages

Printer: 10/10/76 10:10:51 Monitor IPC-0 0070 (100) END

[illegible]