

### TEXT LISTING

068-000313-04

### PROGRAM

DCU-50 (1K OR 4K) MICRO DIAGNOSTIC

### TEXT TAPE

097-000313-04

### ABSTRACT

THIS PROGRAM IS A HARDWARE DIAGNOSTIC FOR THE 4250 OR 4254 DCU-50 PROCESSORS. IT IS DESIGNED TO BE A GATE BY GATE CHECK OF THE PROCESSOR.

COPYRIGHT © DATA GENERAL CORPORATION, 1975,76,78,79  
ALL RIGHTS RESERVED. PRINTED IN U.S.A.

ONLY FOR OPERATION AND MAINTENANCE PURPOSES  
ON DATA GENERAL CORPORATION MANUFACTURED  
EQUIPMENT.

THE AFFIXATION OF A COPYRIGHT NOTICE ON THIS  
DIAGNOSTIC MATERIAL IS NOT INTENDED BY ITSELF  
TO RENDER THE DISTRIBUTION OF THIS DIAGNOSTIC  
MATERIAL A PUBLICATION.

### NOTICE

DATA GENERAL CORPORATION (DGC) HAS PREPARED  
THIS DIAGNOSTIC MATERIAL FOR USE BY DGC  
PERSONNEL AND CUSTOMERS AS A GUIDE TO THE  
PROPER MAINTENANCE OF DGC EQUIPMENT AND  
SOFTWARE. THE DIAGNOSTIC MATERIALS CONTAINED  
HEREIN ARE THE PROPERTY OF DGC AND SHALL  
NEITHER BE REPRODUCED IN WHOLE OR IN PART WITHOUT  
DGC'S PRIOR WRITTEN APPROVAL NOR BE IMPLIED TO  
GRANT ANY LICENSE TO MAKE, USE, OR SELL EQUIPMENT  
OR SOFTWARE MANUFACTURED IN ACCORDANCE HEREWITH.

0001 \*MAIN 10:33:20 10/26/79 10002 \*MAIN  
01  
02  
03  
04  
05  
06  
07  
08  
09  
10  
11  
12  
13  
14  
15  
16  
17  
18  
19  
20  
21  
22  
23  
24  
25  
26  
27  
28  
29  
30  
31  
32  
33  
34  
35  
36  
37  
38  
39  
40  
41  
42  
43  
44  
45  
46

MACRO REV 06.30  
\*\*\*\*\*  
NAME: DCUMD.TX PART NUMBER: 097-000313  
DESCRIPTION: DCU-50 (1K OR 4K) MICRO DIAGNOSTIC  
REVISION HISTORY:  
REV. DATE  
00 09/19/75  
01 04/09/76  
02 07/14/78  
03 12/15/78  
04 10/26/79 NO CHANGES TO TEXT

DATA GENERAL CORPORATION, 1975,1976,1978,1979  
ALL RIGHTS RESERVED  
FOR MAINTENANCE PURPOSES ONLY

THE AFFIXATION OF A COPYRIGHT NOTICE ON THIS  
DIAGNOSTIC MATERIAL IS NOT INTENDED BY ITSELF  
TO RENDER THE DISTRIBUTION OF THIS DIAGNOSTIC  
MATERIAL A PUBLICATION.

NOTICE  
DATA GENERAL CORPORATION (DGC) HAS PREPARED  
THIS DIAGNOSTIC MATERIAL FOR USE BY DGC PER-  
SONNEL AND CUSTOMERS AS A GUIDE TO THE PROPER  
MAINTENANCE OF THE DGC EQUIPMENT AND SOFTWARE.  
THE DIAGNOSTIC MATERIALS CONTAINED HEREIN ARE  
THE PROPERTY OF DGC AND SHALL NEITHER BE RE-  
PRODUCED IN WHOLE OR IN PART WITHOUT DGC'S  
PRIOR WRITTEN APPROVAL NOR BE IMPLIED TO GRANT  
ANY LICENSE TO MAKE, USE, OR SELL EQUIPMENT OR  
SOFTWARE MANUFACTURED IN ACCORDANCE HERewith.

\*\*\*\*\*  
1. PROGRAM NAME: DCUMD\_SR MICRO DIAGNOSTIC FOR  
THE 4250(DCU=50(1K)) OR 4254(DCU=50(4K))

2. REVISION HISTORY:  
01 DTOS STANDARDIZATION  
02 MODIFY FOR DCU=50(4K) TESTING AND  
FOR DTOS/DLIB STANDARDIZATION.  
03 UPDATE FOR DGC, SOFT CONSOLE, DLIB  
CHANGES FOLLOW:  
A. CHANGE ALL NOP TO JMP +1 (FOR AOS ASSEMBLY)  
B. REMOVE ALL OCCURANCES OF READS INSTRUCTION  
C. MOVE ALL MACROS TO BEFORE PAGE ZERO  
D. INSERT P260U FROM DLIB, REMOVE REDUNDANT  
PAGE 0 LABELS  
E. CHANGE EQUATES (=JSR @...) TO USER DEFINED  
SYMBOLS (.DUSR)  
F. INSERT P?STM FROM DLIB, REMOVE FORMER END  
OF PASS ROUTINE  
G. INSERT MESSAGE INDICATING PROGRAM IS LOADED  
AND RUNNING  
H. CHANGE ANY CPU HALTS TO JSR TO ODT (DID NOT  
TOUCH DCU HALTS)  
I. REMOVE "KEYBOARD OR PANEL SWITCHES" QUERY,  
DEFAULT TO KEYBOARD  
J. REPLACE FORMER SETUP, ERROR, LOOP ROUTINES  
WITH DLIB EQUIVALENTS  
K. CHANGE THE DIRT NAME TO 4250 D  
L. MAKE NECESSARY CHANGES IN TEXT FILE TO  
REFLECT ABOVE CHANGES.

3. MACHINE REQUIREMENTS:  
3.1 NOVA/ECLIPSE FAMILY CPU (HOST) WITH AT LEAST  
8K READ/WRITE MEMORY.  
3.2 4250,DCU=50(1K) OR 4254,DCU=50(4K) PROCESSOR  
3.3 4010 TYPE INTERFACE WITH CRT OR TELETYPE  
3.4.0 TEST REQUIREMENTS: N/A

4. FIX PC REPORT PROBLEM ON REV 3  
ADD SPEED UP ROUTINE FOR LOOPX

5. SUMMARY:  
THIS PROGRAM IS A HARDWARE DIAGNOSTIC FOR THE  
4250 OR 4254 DCU=50 PROCESSORS. IT IS DESIGNED TO  
BE A GATE BY GATE CHECK OF THE PROCESSOR.  
THE PROGRAM IS EXECUTED BY THE HOST COMPUTER CONTROLLING  
THE DCU AND ONLY THE HOST COMPUTER IS ASSUMED  
TO BE WORKING PROPERLY.

ALTHOUGH, NO ATTEMPT IS ACTUALLY MADE AT  
DEFINING THE EXACT CAUSE OF ANY TEST FAILURE, THE  
TEST SEQUENCE IS DESIGNED SUCH THAT ONLY A SMALL  
PORTION OF ADDITIONAL LOGIC IS EXERCISED WITH EACH  
SUCCESSIVE TEST. SECTION 7 DESCRIBES THE TEST SEQUENCE  
FOR THIS PROGRAM, AND IT IS HOPED THAT THIS INFORMATION  
MAY BE USED TO ELIMINATE TESTED HARDWARE AS A POSSIBLE  
CAUSE OF TEST FAILURE.

0003 -MAIN

```

01 ; THE INDIVIDUAL TESTS ARE DESCRIBED WITH THE ABILITY
02 ; TO LOOP ON THE CONDITION CAUSING TEST FAILURE,
03 ; THUS PROVIDING A SCOPE LOOP. LOOP OPTIONS ARE DESCRIBED IN
04 ; SECT. 8 AND ERROR PRINTOUTS ARE DESCRIBED IN SECT. 10.
05 ; AS MUCH ADDITIONAL INFORMATION CONCERNING THE TEST
06 ; AND THE NATURE OF THE FAILURE IS CONTAINED IN
07 ; THE LISTING COMMENTS. THE TEST CODE IS DESIGNED TO BE
08 ; EASILY FOLLOWED, PROVIDING THE OPERATOR HAS A BASIC
09 ; KNOWLEDGE OF ASSEMBLY CODE. DEFINITIONS FOR
10 ; SUBROUTINE FUNCTIONS AND RESPECTIVE CALL FORMATS ARE
11 ; CONTAINED IN SECTION 11.
12 ;
13 ;
14 ;
15 ;
16 ;
17 ;
18 ;
19 ;
20 ;
21 ;
22 ;
23 ;
24 ;
25 ;
26 ;
27 ;
28 ;
29 ;
30 ;
31 ;
32 ;
33 ;
34 ;
35 ;
36 ;
37 ;
38 ;
39 ;
40 ;
41 ;
42 ;
43 ;
44 ;
45 ;
46 ;
47 ;
48 ;
49 ;
50 ;
51 ;
52 ;
53 ;
54 ;
55 ;
56 ;
57 ;
58 ;
59 ;
60 ;

```

10004 -MAIN

```

01 ;
02 ;
03 ;
04 ;
05 ;
06 ;
07 ;
08 ;
09 ;
10 ;
11 ;
12 ;
13 ;
14 ;
15 ;
16 ;
17 ;
18 ;
19 ;
20 ;
21 ;
22 ;
23 ;
24 ;
25 ;
26 ;
27 ;
28 ;
29 ;
30 ;
31 ;
32 ;
33 ;
34 ;
35 ;
36 ;
37 ;
38 ;
39 ;
40 ;
41 ;
42 ;
43 ;
44 ;
45 ;
46 ;
47 ;
48 ;
49 ;
50 ;
51 ;
52 ;
53 ;
54 ;
55 ;
56 ;
57 ;
58 ;
59 ;
60 ;

```

PROGRAM DESCRIPTION

THE PROGRAM IS BASICALLY SPLIT UP INTO TWO PARTS. IN THE 1ST PART THE HOST EXECUTES THE DCU IN DIAGNOSTIC MODE IN WHICH THE DCU-CPU CLOCK IS DISABLED AND ALL TIMING IS CONTROLLED VIA THE HOST I/O PLS. (IE. ONE I/O PLS FROM THE HOST PRESENTS 1 PULSE BY THE DCU 10 MHZ CLOCK). THE 2ND PART EXECUTES THE DCU UNDER NORMAL MODE IN WHICH THE DCU IS SEQUENCED BY ITS 10 MHZ CLOCK.

BOTH PARTS ARE MORE FULLY EXPLAINED AS FOLLOWS:

DIAGNOSTIC MODE

WITH THE DCU UNDER TOTAL CONTROL OF THE HOST, TWO HOST TO DCU I/O INSTRUCTION BECOME OF SPECIAL IMPORTANCE.

DIA N,SCP AND  
DIB N,SCP

THE 1ST (DIA) MONITORS THE 16 SIGNALS DESCRIBED BELOW

BIT0 - SETCARRY - BECOMES CARRY AFTER NEXT CPUCK  
BIT1 - SETSKIP (INV) - BECOMES SKIP AFTER NEXT CPUCK  
BIT2 - PULSE (IE. 9,C,P)  
BIT3 - DEFER (INV) - I/O PULSE  
BIT4 - CPUCK - INDIRECT CYCLE  
BIT5 - SC WRITE - BASIC CPU CYCLE CLOCK  
BIT6 - DP2 - ACTUAL WRITE ENABLE TO SC MEM  
BIT7 - DPI - LDA/STA ETC EXECUTE STATE  
BIT8 - DP1 - EFA/ALC/ I/O STATE  
BIT9 - 1RADR0 - ADDRESS LINES TO REGISTER FILES  
BIT10 - 1RADR2  
BIT11 - 2HRADR0  
BIT12 - 2HRADR1  
BIT13 - 1WADR0  
BIT14 - 1WADR1  
BIT15 - COMADR2

BY MAINTAINING CONTROL OVER THE DCU PROGRESS THROUGH A KNOWN INSTRUCTION SEQUENCE VIA THE I/O PLS AND MONITORING THE DIA WORD, MANY OF THE CPU CONTROL PROBLEMS MAY BE DETECTED IMMEDIATELY IN THE PARTICULAR MACHINE STATE IN WHICH THEY OCCUR, OR IN THE FOLLOWING STATE IN THE CASE OF DEFER, DPI, AND DP2

THE 2ND I/O FUNCTION DIB N,SCP MONITORS THE DCU MADR LINES. IN THE HALT STATE THESE LINES REFLECT THE DCU PC. BUT DEPENDING ON THE MACHINE STATE THEY MAY REFLECT THE MA REG OR ANY OF THE 4 ACCUMULATORS.

A SUBROUTINE PACKAGE EXISTS TO CONTROL THE STARTING, SEQUENCING, AND DIA, DIB CHECKING OF THE DCU THROUGH VARIOUS INSTRUCTION SEQUENCES, ALSO CONTAINED IN THE PROGRAM.

0005 .MAIN

10006 .MAIN

```

01 ; 7.2 NORMAL MODE
02 ; WITH THE DCU CPU ACTUALLY RUNNING WITH
03 ; ITS OWN 10 MHZ CLOCK, THE HOST PROGRAM
04 ; ESSENTIALLY STARTS THE DCU AT A PARTICULAR
05 ; STARTING ADDRESS, WAITS FOR IT TO HALT, AND
06 ; CHECKS THAT THE FINAL PC IS CORRECT.
07 ;
08 ; A CALL TO DMHLT WILL CAUSE THE DCU AC'S TO
09 ; BE TRANSFERRED TO THE HOST AC'S AND PRINTED
10 ; VIA THE EMAL ROUTINE.
11 ;
12 ; THE MAIN OBJECTIVES OF THIS PART OF THE PROGRAM
13 ; ARE:
14 ;
15 ; 1. TO CHECK THAT THE MUCH HIGHER CLOCK FREQUENCY
16 ; PRESENTS NO TIMING PROBLEMS.
17 ;
18 ; 2. TO CHECK OPERATION OF THE 1K DCU SC MEMORY.
19 ;
20 ; 3. TO CHECK OUT THE I/O INTERFACE/INTERRUPT LOGIC.
21 ;
22 ; 7.3 PROGRAM NOTES
23 ; PROGRAM DOES NOT CHECK ANY OF THE SLAVE (DCU)
24 ; I/O CONTROL OR DATA LINES EXCEPT THOSE WHICH ARE
25 ; USED WITH THE HOST/DCU INTERFACE.
26 ; SIGNALS CHECKED ARE SLAVE(BUSY, DONE, START, PULSE,
27 ; CLEAR, FORST), DDA N, SPU(BIT 15 -ONLY)
28 ; MSKO (BIT 4=ONLY) AND DIA N, SPU (READ HTS REGISTER)
29 ;
30 ; *****
31 ; 7.4 PROGRAM LAYOUT (LISTING)
32 ;
33 ; LOC'S 0-6431 = HOST EXECUTED CODE (MAIN PROGRAM) TO
34 ; CONTROL AND MONITOR THE DCU EXECUTING THE CODE
35 ; FOUND AT LOC'S 10000-13530.
36 ;
37 ; *****
38 ; LOC'S 6432-6601 = HOST EXECUTED SUBROUTINES
39 ; SUGGEST READING ABOVE MENTIONED PAGES 1ST FOR
40 ; QUICKER PROGRAM UNDERSTANDING
41 ; *****
42 ;
43 ; LOC'S 6602-7231 = DIAGNOSTIC (DIA N, SCP) CHECKWORDS
44 ;
45 ; LOC'S 10000-13530 = DCU EXECUTED CODE
46 ;
47 ; LOC'S 13531-END = INITIALIZATION, ERROR HANDLING
48 ; TIMING, AND TTY PACKAGE, SWITCH PACK, OCTAL DEBUG TOOL.
49 ;

```

```

; 8.
; SWITCH SETTINGS
; LOCATION "SWREG" IS USED TO SELECT THE PROGRAM OPTIONS
; (NOT SYSTEM CONFIGURATION). WHILE RUNNING UNDER DTOS,
; THIS LOCATION WILL BE LOADED BY THE MONITOR.
; UNDER STAND ALONE AND PROGRAM LOAD MODES THIS
; LOCATION WILL BE SET ACCORDING TO THE ANSWERS SUPPLIED
; BY THE OPERATOR. IN ANY CASE, THE OPTIONS CAN BE CHANGED
; OR VERIFIED BY USING ONE OF THE COMMANDS GIVEN IN SEC.
; 8.2
;
; SWITCH OPTIONS
; DIFFERENT BITS AND THEIR INTERPRETATION AT LOCATION
; "SWREG" IS AS FOLLOWS:
;
; BIT OCTAL BINARY INTERPRETATION
; VALUE VALUE
;
; 1 40000 1 0 LOOP ON ERROR
; SKIP LOOPING ON ERROR
;
; 2 20000 1 0 PRINT TO CONSOLE
; ABORT PRINT OUT TO CONSOLE
;
; 3 10000 1 0 DO NOT PRINT % FAILURE
; PRINT % FAILURE
;
; 4 04000 1 0 ALLOW END OF PASS PRINT OUT
; SUPPRESS END OF PASS PRINT OUT
;
; 5 02000 1 0 DO NOT PRINT ON THE LINE PRINTER
; PRINT ON THE LINE PRINTER
;
; 6 01000 1 0 DO NOT HALT ON ERROR
; HALT ON ERROR
;
; 7 00400 1 0 DON'T PRINT SUBTEST SUMMARY OR PASS
; PRINT SUMMARY AND/OR
; PASSING OF EACH SUBTEST
; PRINT ONLY THE FIRST ERROR
; PRINT EVERY ERROR
;
; 8 00200 1 0
;
; 8.2
; SWITCH COMMANDS
; ONCE THE PROGRAM STARTS EXECUTION THE STATE OF ANY OF
; THE BITS CAN BE CHANGED BY HITTING KEYS 1-9, A-F. THE
; PROGRAM WILL CONTINUE RUNNING AFTER UPDATING THE OPTIONS.
; EACH KEY WILL COMPLEMENT THE STATE OF THE BIT AFFILIAT-
; ED WITH IT, THUS BIT 4 CAN BE ALTERED BY HITTING KEY 4.
; SETTING OF ANY BIT OF LOCATION "SWREG" WILL SET BIT 0.
; (DEFAULT MODE IS DEFINED AS ALL BITS OF SWREG SET TO 0)
; THE PROGRAM CAN BE LOCKED INTO SWITCH MODIFICATION MODE
; BY TYPING A 0, IN WHICH CASE MORE THAN ONE BIT CAN BE
; CHANGED BEFORE CONTROL IS ALLOWED TO RETURN TO THE
; MAIN PROGRAM.
;
; 8.2.1 OTHER COMMANDS
;

```

0007 .MAIN

```

01
02
03
04
05
06
07
08
09
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26

```

"CR" A "RETURN" CAN BE TYPED TO CONTINUE THE PROGRAM AFTER ITS LOCKED IN A SWITCH MODIFICATION MODE

"D" THIS COMMAND GIVEN AT ANY TIME WILL RESET "SMREG" TO DEFAULT MODE AND RESTART THE PROGRAM.

"R" THIS COMMAND GIVEN AT ANY TIME WILL RESTART THE PROGRAM. SWITCHES ARE LEFT WITH THE VALUES THEY HAD BEFORE THE COMMAND WAS ISSUED.

"O" THIS COMMAND GIVEN AT ANY TIME WILL CAUSE THE PROGRAM CONTROL TO GO TO OOT (NOTE: THIS IS AN OPTIONAL COMMAND AND IS AVAILBLE ONLY IF OOTPK IS PRESENT)

M THIS COMMAND GIVEN AT ANY TIME WILL PRINT THE CURRENT OPERATING MODES.

18.3 OTHER SWITCH OPTIONS

SW 15 = 0 DO NOT USE DELAY IN LOOP ROUTINE

SW 15 = 1 USE DELAY IN LOOP ROUTINE (SEE 9.4 BELOW)

0008 .MAIN

```

01
02
03
04
05
06
07
08
09
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60

```

9. OPERATING PROCEEDURE/OPERATOR INPUT:

- LOAD USING THE BINARY LOADER OR DTOS/DOOS MEDIUM.
- STARTING ADDRESSES
  - 200- START DIAGNOSTIC
  - 240- START ADDRESS FOR STARTING ANY OPERATOR DEFINED SCOPE LOOP STARTING AT ADDRESS TOP.
- THE OPERATOR IS ASKED FOR THE DCU DEVICE CODE WHICH MAY BE ANY OCTAL INPUT 1-76, AND WHETHER THE DCU IS OF THE 1K (DCU=50) OR THE 4K (DCU= ) DESIGN, TO WHICH THE RESPONSE SHOULD BE EITHER 1 OR 4.
- DUE TO THE USE OF THE IORST WITHIN THE TEST LOOPS TO RESET THE DCU IN ITS DIAGNOSTIC MODE, THE PROGRAM MAY NOT RESPOND TO ANY KEYBOARD INPUTS DURING TEST EXECUTION.
  - TO ENABLE SWITCH INPUTS VIA THE KEYBOARD, USE THE FOLLOWING PROCEDURE:(ONLY AFTER ERROR)
    - SET BIT 6 (HALT ON ERROR) AT THE BEGINNING OF THE PROGRAM DURING OPERATOR INPUT;
    - UPON ERROR, THE CPU WILL HALT AT THIS TIME, INPUT THE DESIRED SWITCH VIA THE KEYBOARD;
    - HIT "CONTINUE" ON THE FRONT PANEL. THE CPU WILL HALT AGAIN;
    - REPEAT STEPS B AND C UNTIL ALL DESIRED SWITCHES HAVE BEEN INPUTTED;
    - WHEN DONE INPUTTING SWITCHES, FINALLY INPUT SWITCH 6 (CLEARING SW 6);
    - HIT "CONTINUE" AND THE PROGRAM WILL CONTINUE, WITH THE NEW SWITCH DATA.

THE ONLY RESTRICTION TO THIS TECHNIQUE IS THAT SWITCHES MAY ONLY BE INPUT AFTER AN ERROR HAS OCCURRED, AND ONCE DONE, CAN NOT BE DONE AGAIN WITHOUT MANUALLY HALTING THE CPU AND RESTARTING THE PROGRAM.

AN ALTERNATE METHOD IS AVAILABLE WHICH WILL ALLOW KEYBOARD INPUT AT ANY TIME, BUT WILL APPRECIABLY INCREASE THE PROGRAM RUN TIME AS WELL AS LENGTHEN THE TEST LOOPS. (THIS METHOD SHOULD ONLY BE USED IF KEYBOARD INPUT IS A NECESSITY.)

A. ONCE THE PROGRAM IS LOADED AND STARTS ASKING QUESTIONS, IT WILL ASK FOR SWITCHES TO BE SET. IF KEYBOARD ENTRY IS DESIRED DURING TEST EXECUTION, SET SWITCH 15 (VIA THE KEYBOARD) ALONG WITH ANY OTHER DESIRED SWITCH OPTIONS. THIS WILL ALLOW SWITCH INPUTS DURING EXECUTION BUT WILL GREATLY INCREASE PROGRAM RUN TIME. ONCE THE DELAY HAS BEEN SET, IT CAN BE CLEARED IN THE FOLLOWING MANNERS:

- INPUT "D" TO CLEAR ALL SWITCHES AND RESTART THE PROGRAM;
- INPUT "R" TO RESTART THE PROGRAM. WHEN ASKED TO SET SWITCHES, DO NOT SET 15;

0009 \*MAIN

```
01 ?  
02 ?  
03 ?  
04 ?  
05 ?  
06 ?  
07 ?  
08 ?  
09 ?  
C. INPUT *0 TO ENTER ODT, CHANGE LOCATION  
ERR24 (211) FROM -1 TO 1. THIS WILL  
REMOVE THE DELAY TIME FROM THE LOOP  
ROUTINE. THEN INPUT P TO RETURN FROM ODT  
AND CONTINUE THE PROGRAM.  
5. PROGRAM PRINTS PASS FOLLOWING THE 1ST COMPLETION  
OF TESTS AND AFTER EVERY 20 TH COMPLETION  
THEREAFTER.
```

10010 \*MAIN

```
01 ?  
02 ?  
03 ?  
04 ?  
05 ?  
06 ?  
07 ?  
08 ?  
09 ?  
10 ?  
11 ?  
12 ?  
13 ?  
14 ?  
15 ?  
16 ?  
17 ?  
18 ?  
19 ?  
20 ?  
21 ?  
22 ?  
23 ?  
24 ?  
25 ?  
26 ?  
27 ?  
28 ?  
29 ?  
30 ?  
31 ?  
32 ?  
33 ?  
34 ?  
35 ?  
36 ?  
37 ?  
38 ?  
39 ?  
40 ?  
41 ?  
42 ?  
43 ?  
44 ?  
45 ?  
46 ?  
47 ?  
48 ?  
49 ?  
50 ?  
10. PROGRAM OUTPUT/ERROR DESCRIPTION  
WHEN AN ERROR IS DETECTED A VALUE FOR AN  
ERROR PC WILL BE PRINTED ALONG WITH AC'S 0,1 AND 2  
OF THE HOST COMPUTER.  
THE PC VALUE PRINTED MAY POINT TO ONE OF THE  
FOLLOWING:  
1. A CALL TO EHMT  
2. A CALL TO THE ROUTINE WHICH DETECTED  
THE ERROR AND SUBSEQUENTLY EXITED  
TO EHMT.  
3. A HALT IN AN ACTUAL INSTRUCTION SEQUENCE  
BEING EXECUTED BY THE DCU.# FOUND  
IN LAST PORTION OF THE PROGRAM IN WHICH  
THE DCU IS BEING EXECUTED IN NORMAL MODE  
THE CALL DMHLT SETS UP THIS CONDITION.  
THE SIGNIFICANCE OF AC'S 0&2 IS EXPLAINED IN THE  
LISTING AT THE SOURCE OF THE ERROR. IN GENERAL  
ACO CONTAINS THE CHECKWORD AND AC1 THE DATA IN  
QUESTION.  
THE CALL DMHLT WILL CAUSE THE DCU AC'S TO BE  
PRINTED, WHILE IN ALL OTHER CASES THE AC'S  
PRINTED WILL BE THOSE OF THE HOST  
*****  
# ANY ADDRESSES REFERRING TO LOC'S 400-1400  
REFER TO LISTING FOR TESTS N7-N10 AND ASSOCIATED  
CODE T107,8,9 AND T  
*****  
# IF THE PROGRAM HAS DETECTED AN ERROR WHILE  
EXERCISING THE DCU IN ITS DIAGNOSTIC MODE,  
THE DCU REMAINS IN THE PARTICULAR MACHINE STATE  
THAT THE ERROR WAS DETECTED IN UNTIL THE OPERATOR  
HITS CONTINUE.  
AFTER PRINTING THE ABOVE THE PROGRAM HALTS ALLOWING  
THE OPERATOR TO SET THE SWITCHES. UPON PRESSING  
CONTINUE THE PROGRAM MAY EITHER GO ON TO THE NEXT  
TEST OR GO INTO A SCOPE LOOP BETWEEN THE APPROPRIATE  
CALLS TO SETUP AND LOOP (SEE SECTION 8. SWITCH SETTINGS)  
IF AN UNEXPECTED INTERRUPT OCCURS, THE PROGRAM WILL  
INDICATE THE INTERRUPTING DEVICE CODE AND EXIT TO  
ODT. CONTINUING FROM ODT WILL CLEAR THE INTERRUPTING  
DEVICE AND RETURN TO PROGRAM EXECUTION. (SEE 11B BELOW  
FOR ODT COMMANDS).
```

```

10011 .MAIN
01
02
03
04
05
06
07
08
09
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
;11. DEBUG HELP
;11A.0 SUBROUTINES AND CALL FORMATS
;HOST EXECUTED
;DCU DIAGNOSTIC SUBROUTINE PACKAGE
;TIMING AND START UP ROUTINES
;*****
;CALL PULSE GENERATE *****
; N NO. OF I/O PULSES *****
;CALL CLKZH *****
; N1 # OF PULSES TO SET CPUCK L *****
; N2 # OF PULSES TO THEN SET CPUCK H *****
; NORMAL RETURN ~ N1+N2 EXCEEDED *****
;*****
;CALL CPUCK *****
; N # OF CPUCK CYCLES *****
;*****
;CALL SRTUP START DCU IN DIAGNOSTIC MODE *****
; ADDR STARTING ADDRESS *****
;CALL SRTN START DCU IN NORMAL MODE *****
; ADDR STARTING ADDRESS *****
;*****
;CALL HWAIT *****
; N # OF LOOP PASSES UNTIL TIMEOUT *****
; APPR. 2-3 US PER N *****
;*****
;CALL DMHLT XFER DCU AC'S TO HOST AC'S, AND *****
; DCU HALT ADDRESS TO ERROR PC(.RTRN) *****
;*****
;CALL SZWT WAIT UNTIL BUSY=0 (2 SEC TIMEOUT)

```

```

10012 .MAIN
01
02
03
04
05
06
07
08
09
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
;UTILITY ROUTINES
;*****
;CALL ADDR ADDRESS OF PROPER CALL TO LOOP *****
; LADDR *****
;CALL COMP COMPARE (A1) TO (A2) *****
; A1 *****
; A2 *****
; ERROR RETURN A1 NOT = A2 *****
; NORMAL RETURN MATCH *****
;*****
;DIAGNOSTIC CHECK ROUTINES
;*****
;CALL COAD CHECK DIA AGAINST (CADD) *****
; CADD *****
; ERROR RETURN *****
; NORMAL RETURN *****
;*****
;CALL COIA CHECK DIA AGAINST (CADD) *****
; CADD BUT IGNORE CARRY *****
; ERROR RETURN *****
; NORMAL RETURN *****
;*****
;CALL COIB CHECK DIB AGAINST CX *****
; CX *****
; ERROR RETURN *****
; NORMAL RETURN *****
;*****
;CLOCK AND CHECK ROUTINES
;*****
;CALL EX(1-4)M OR EXIO *****
; CADD *****
; NORMAL RETURN *****
;*****
;EXECUTE I=4 CPUCK CYCLES/ AND CHECK DIA AGAINST *****
;(CADD) IF AN ERROR IS DETECTED EXIT TO EHMT+1 *****
;PC PRINTED UPON JMP TO EHMT+1 WILL REPRESENT CALL *****
; TO EX(1-4)M.
;*****
;CALL COIBC CHECK DIB AGAINST CX *****
; CX *****
; NORMAL *****
;*****
;ERROR EXIT - SEE EXPLANATION FOR EX(1-4)M

```

10013 .MAIN

```

01 ;MULTIPLE CHECK/CLOCK ROUTINES USED IN EXECUTING
02 ;SEQUENCES OF THE FORM
03   LDA 0,TC10-1
04   LDA 1,TC10-2
05   ALC 0,1
06   STA 1,TC10-4
07   HALT
08 ;WHERE THE OPERATION ALC 0,1 IS IN QUESTION
09 ;UPON FINDING AN ERROR THE ROUTINE EXITS TO EHALT*1
10 ;AND THE CALL TO .LLFS IS USED AS ERROR PC. SINCE
11 ;THERE ARE MORE THAN 1 CHECK, THE ONE WHICH
12 ;FAILED MAY BE DETERMINED BY EXAMINING THE
13 ;CHECKWORD IN AC0
14
15 ;CALL EXLLFS EXECUTE SERIES AT ADDRESS TADD
16   TADD
17 ;
18 ;NORMAL RETURN
19 *****
19 ;CALL EXLLSS EXECUTE SERIES AT
20   TADD
21 ;
22 ;NORMAL RETURN
23
24 ;SAME AS EXLLFS EXCEPT REGISTER 3 AND 0 ARE USED AND
25 ;THE ALC BECOMES ALC 3,0,3,0
26 ;SEE EXLLFS TEST DESCRIPTION
27 *****
28
29
30 ;SETUP DATA BLOCK ROUTINES
31 ;SETUP 4 WORD BLOCK AS FOLLOWS
32
33 ;
34 ; RESULT
35 ; CHECKWORD OR 2ND RESULT
36 ; OPERAND 2
37 ; OPERAND 1
38 ; TADD:
39 *****
40 ;CALL MRST
41 ; TADD DATA BLK ADDRESS
42 ; CADD CONSTANT ADDRESS (2)
43 *****
44 ;CALL ADST
45 ; TADD DATA BLK ADDRESS
46 ; OP1
47 ; OP2
48 ; RESULT CHECKWORD
49 *****
50 ;CALL MVST
51 ; TADD DATA BLK ADDRESS
52 ; OP1
53 ; RESULT CHECKWORD
54 *****
55 ;CALL CLR4 CLEAR 4 WORDS
56 *****

```

10014 .MAIN

```

01
02
03
04
05
06
07
08
09
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60

```

OCTAL DEBUG TOOL (ODT)  
 THIS DIAGNOSTIC IS EQUIPPED WITH A BUILT IN ODT WHICH CAN BE ACCESSED BY HITTING CONTROL 0 ("O") AT ANY TIME DURING THE EXECUTION OF THE PROGRAM (AFTER SETTING THE PARAMETERS).  
 ON ENTERING ODT THE ADDRESS OF THE LOCATION HAVING THE NEXT INSTRUCTION TO BE EXECUTED WILL BE TYPED-OUT.  
 CONVENTIONS AND SYMBOLS  
 THE FOLLOWING CONVENTIONS ARE USED BY THE ODT:  
 ? POUND WITH A "3"  
 @ ODT IS READY AND AT YOUR SERVICE.  
 COMMAND STRUCTURE  
 AN ODT COMMAND HAS THE FOLLOWING FORMAT:  
 [ARGUMENT] [COMMAND]  
 AN ARGUMENT MAY BE ONE OF THE FOLLOWING:  
 "EXP" AN OCTAL EXPRESSION CONSISTING OF OCTAL NUMBERS SEPARATED BY PLUS (+) OR MINUS (-) SIGNS. LEADING ZEROS NEED NOT BE TYPED.  
 "ADR" AN ADDRESS IS THE SAME AS AN EXPRESSION EXCEPT THAT BIT 0 IS NEGLECTED.  
 A COMMAND IS A SINGLE TELETYPE CHARACTER  
 ODT COMMANDS  
 THE LOCATIONS THAT CAN BE EXAMINED AND MODIFIED BY THE USER ARE CALLED CELLS. THESE CELLS ARE OF TWO TYPES:  
 INTERNAL CPU CELLS AND MEMORY LOCATIONS.  
 OPENING INTERNAL CELLS  
 THE COMMAND TO OPEN ONE OF THE INTERNAL REGISTERS IS OF THE FORM "NA" WHERE N IS ANY OCTAL EXPRESSION BETWEEN 0 AND 7  
 0-3 FOR ACCUMULATORS 0-3  
 4 FOR PC OF THE NEXT INSTRUCTION TO BE EXECUTED IN THE EVENT OF A "P" COMMAND.  
 5 CPU AND I/O STATUS  
 BIT INTERPRETATION  
 15 STATUS OF I/O DONE FLAG  
 14 STATUS OF INTERRUPTS (ION FLAG)  
 13 STATUS OF CARRY BIT  
 6 ADDRESS OF THE LOCATION HAVING THE BREAK POINT (IF ANY)  
 7 INSTRUCTION AT THE BREAK POINT LOCATION  
 OTHER COMMANDS TO OPEN CELLS ARE:  
 "ADR"/ OPEN THE CELL AND PRINT ITS CONTENTS  
 ./ OPEN THE CELL CURRENTLY POINTED TO BY THE POINTER AND PRINT ITS CONTENTS.  
 ."ADR"/ ADD "ADR" TO THE POINTER, OPEN THE CELL  
 ."ADR"/ AND PRINT ITS CONTENTS.  
 ."ADR"/ FROM THE POINTER, OPEN THE CELL AND PRINT ITS CONTENTS.  
 "CR" THE RETURN KEY IS USED TO CLOSE THE OPEN CELL WITH OR WITHOUT MODIFICATION.



```

0015 .MAIN
01
02
03
04
05
06
07
08
09
10
11
12
10016 .MAIN
01
02
03
04
05
06
07
08
09
10
11
12

```

```

"LF" LINE FEED IS USED TO CLOSE THE OPEN CELL WITH OR
WITHOUT MODIFICATION AND TO OPEN THE SUCCEEDING
CELL.
" " CLOSE THE OPEN CELL WITH OR WITHOUT MODIFICATION
AND OPEN THE PRECEDING CELL
/ CLOSE THE OPEN CELL WITHOUT MODIFICATION, AND
OPEN THE CELL POINTED TO BY ITS CONTENTS.
+"ADR"/ CLOSE THE OPEN CELL WITHOUT MODIFICATION, AND
OPEN THE CELL POINTED TO BY ITS CONTENTS + "ADR".
-"ADR"/ CLOSE THE OPEN CELL WITHOUT MODIFICATION, AND
OPEN THE CELL POINTED TO BY ITS CONTENTS - "ADR".

```

```

118.3.2 MODIFICATION OF A CELL
ONCE A CELL HAS BEEN OPENED ITS CONTENTS CAN BE MODIFIED
BY TYPING THE NEW VALUE THE CELL IS TO CONTAIN IN THE
FORM OF AN OCTAL EXPRESSION FOLLOWED BY "CR" OR "LF".
IF A + OR - IS TYPED AS THE FIRST CHARACTER OF THE EX-
PRESSION THEN THE VALUE OF THE EXPRESSION IS ADDED TO OR
SUBTRACTED FROM THE OLD CONTENTS OF THE CELL. THE
ADDRESS ITSELF OR AN EXPRESSION RELATIVE TO THE ADDRESS
CAN BE DEPOSITED BY TYPING A ".", "OR" ".%"/OCTAL EXPRESS-
ION". A RUBOUT COMMAND GIVEN RIGHT AFTER OPENING A CELL
ALLOWS THE MODIFICATION OF ITS CONTENTS AS IF THEY WERE
TYPED IN JUST BEFORE THE COMMAND WAS ISSUED.

```

```

118.3.3 OTHER ODT COMMANDS
THIS KEY IS USED TO DELETE ERRONEOUSLY TYPED
DIGITS. EACH TIME THE KEY IS PRESSED THE RIGHT MOST
DIGIT IS DELETED AND ECHOED ON THE TERMINAL. IF
THE RUBOUT KEYS PRESSED RIGHT AFTER OPENING A
CELL THEN IT DELETES THE RIGHT MOST DIGIT OF THE CELLS
CONTENTS. THIS ALLOWS THE MODIFICATION OF THE CELL
AS IF ITS CONTENTS WERE TYPED IN JUST BEFORE THE
KEY WAS PRESSED.
"ADR"B INSERT A BREAK POINT AT LOCATION "ADR".
ONLY ONE BREAK POINT CAN BE INSERTED AND ANY
ENTRY TO ODT AFTER EXECUTING A BREAK POINT WILL
CAUSE IT TO BE DELETED.
D DELETE THE BREAK POINT IF ANY.
P RESTART THE EXECUTION OF THE PROGRAM AT LOCATION
POINTED BY 44.
"ADR" START EXECUTING THE PROGRAM AT "ADR" AFTER AN
IO-RESET.
K KILL THE STRING TYPED SO FAR. THE ODT RESPONDS
WITH A "?" AND THE OPEN CELL IS CLOSED WITHOUT
MODIFICATION.
= PRINT THE OCTAL VALUE OF THE INPUT ONLY.
THIS WILL CLOSE ANY OPEN CELLS WITHOUT
MODIFICATION AND WILL NOT OPEN A CELL

```

```

NOTE: IN PROGRAMS WHICH RELOCATE THEMSELVES THE
THE USER SHOULD PLACE BREAK POINTS ONLY IN THE
ORIGINAL PROGRAM AREA. IF A BREAK POINT IS
PLACED OUTSIDE THIS AREA THE RESULTS WILL
BE UNPREDICTABLE.

```

```

112.0 SPECIAL NOTES:
PROGRAM DOES NOT CHECK ANY OF THE SLAVE (DCU)
I/O CONTROL OR DATA LINES EXCEPT THOSE WHICH ARE
USED WITH THE HOST/DCU INTERFACE. SIGNALS CHECKED
ARE SLAVE(BUSY,DONE,START,PULSE,CLEAR,IORST),
DIA N,SPU(BIT 15 ONLY), MSKO (BIT 4 ONLY) AND
DIA N,SPU (READ HTS REGISTER)
113.0 RUN TIME:
N/A
.EOT

```

0017 .MAIN

\*\*00000 TOTAL ERRORS, 00000 PASS 1 ERRORS

0018 .MAIN

02DTD 001514 MC 14/01  
S7WPD 001044 MC 6/01