

**DataGeneral**

---

---

**TECHNICAL  
STATEMENT**

---

---

TEXT LISTING

068-001074-01

PROGRAM

ARRAY PROCESSOR EXERCISER-J

TEXT TAPE

097-001074-01

ABSTRACT

THIS PROGRAM IS A FUNCTIONAL TEST FOR THE ARRAY PROCESSOR (AP).  
IT IS EXECUTED BY THE CENTRAL PROCESSOR (OR IOP) CONTROLLING THE  
AP AND TESTS SPECIFIC AP INSTRUCTIONS.

COPYRIGHT © DATA GENERAL CORPORATION, 1978  
ALL RIGHTS RESERVED. PRINTED IN U.S.A.



10003 APJ.T

```

01 PROGRAM DESCRIPTION/THEORY OF OPERATION
02 APJ TEST DESCRIPTION
03
04 THE APJ EXERCISER VERIFIES
05
06 INR = DIGITAL INTEGRATION
07 NRA = NEGATE A REAL ARRAY
08 LDC = LOAD A COMPLEX ARRAY
09 STC = STORE A COMPLEX ARRAY
10 LDRP = LOAD A REAL ARRAY (PARTIAL LOAD)
11 ARSP = ADD A SCALAR TO REAL ARRAY (PARTIAL LOAD)
12
13 INR IS TESTED USING A SIMULATION APPROACH.
14 THE INSTRUCTION IS TESTED IN SEVERAL WAYS
15 WITH SEQUENTIAL FLOATING POINT DATA AND THEN
16 RANDOM FLOATING POINT DATA.
17
18 A) STRUCTURE #1 TEST INR FOR N FROM 2 -> 2000,
19 IJ=1, IK=1.
20
21 2000(8) ELEMENTS OF DATA AT FDATA ARE LOADED TO
22 START OF AP RAM. THE INSTRUCTION IS SIMULATED
23 FOR N=2000. SIMULATED RESULTS GO TO
24 GDATA BUFFER.
25
26 THEAP EXECUTES INR. AP RESULTS GO TO AP RAM
27 LOC. STARTING AT 2000 (2NM) THE TWO OUTPUT
28 ARRAYS ARE COMPARED VIA RCOMP ROUTINE.
29
30 B) THE SECOND STRUCTURE WILL CHECK OUT INPUT AND
31 OUTPUT STARTING ADDRESS VARIATIONS CAPABILITIES.
32 VARIOUS VALUES OF N ARE ALSO USED HERE.
33
34 THIS IS DONE VIA RANDOM SLICING OF INPUT DATA.
35 OUTPUT DATA WILL ALSO GO TO SOME RANDOM SLICE
36 IN UPPER HALF OF AP RAM.
37
38 THE CONDITIONS ARE SIMULATED WITH RESULTS GOING
39 TO GDATA BUFFER.
40 RESULTS ARE COMPARED VIA RCOMP. THIS RANDOM
41 SLICE PROCESS IS EXECUTED 200(8) TIMES.
42
43 C) TEST STRUCTURES 3A AND 3B ARE SIMILAR TO
44 PREVIOUS TESTS IN THAT SIMULATION AND COMPARISON
45 IS THE METHOD USED.
46 HERE N IS FIXED AT 101. IJ=10, IK=1. THIS IS
47 SIMULATED AND THEN COMPARED TO ACTUAL RESULTS.
48
49 3B REVERSES INDEJNG IJ=1, IK=10.
50
51 THIS STRUCTURE CHECKS OUT IF INDEJNG IS IN CORRECT
52 DIRECTION.

```

10004 APJ.T

```

01
02 D) THE LAST STRUCTURE (#4A)
03
04 USES RANDOM N, IJ, IK
05 N VARIES FROM 1-100(8) RANDOMLY
06 IJ VARIES FROM 1-17(8) RANDOMLY
07 IK VARIES FROM 1-17(8) RANDOMLY
08
09 FOR EACH SET OF RANDOM PARAMETERS, THE INSTRUCTION
10 IS SIMULATED AND THEN EXECUTED. THE RESULTS ARE
11 COMPARED. GDATA BUFFER HAS SIMULATED RESULTS WHILE
12 AN ARRAY STARTING AT LOC 2000 (2NM AP RAM) CONTAINS
13 ACTUAL RESULTS.
14 THIS PROCEDURE IS DONE 1000(8) TIMES.
15
16 STRUCTURES 1,2,3A,3B,4A ARE REPEATED OVER AGAIN USING
17 A RANDOM FLOATING POINT DATA SET AT FDATA. THIS WILL
18 INSURE THAT THE INR IS CHECKED FOR DATA SENSITIVITY.
19
20 7.1.2 LDC AND STC ARE TESTED TOGETHER. THE TESTING IS QUITE
21 SIMPLE. IN ALL CASES, THE DATA IN BUFFER FDATA IS
22 A BIT PATTERN. THE DATA IS LOADED TO RAM USING LOC.
23 THIS LOAD IS CHECKED USING RCOMP ROUTINE. THE DATA
24 IS THEN STORED BACK TO MM AT IDATA BUFFER. THE MM
25 DATA BUFFERS IDATA AND FDATA ARE THEN CHECKED AGAINST
26 EACH OTHER.
27
28 A) TEST #1 VARIES N => 1 -> 512.
29 CHECKS IF N IS FUNCTIONAL
30 B) TEST #2, N = 50. ISOURCE = 1.
31 IOEST = 10.
32 C) TEST #3, N = 50. IOEST = 10.
33
34 BOTH 2 AND 3 CHECK IF INDEJNG IS WORKING OK.
35
36 D) TEST #4 IS A RANDOM SLICE TEST. THIS CHECKS
37 TO SEE IF THE STC AND CDC CAN HANDLE VARIOUS AP
38 RAM START ADDRESS PLUS VARIOUS VALUES OF N.
39
40 E) TEST #5 VARIES N, IM, IZ ALL RANDOMLY
41 N => 1-101.
42 IM, IZ 1-10.
43
44 :NOTE: ALL THE TESTS FOLLOW THE SAME BASIC STRUCTURE.
45 INPUT DATA AT FDATA. OUTPUT BUFFER AT IDATA.
46 IDATA IS CHECKED AGAINST FDATA.

```



10007 APJ.T

```

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

```

C) THE NEXT TWO TESTS CHECK OUT IF SOURCE AND DEST INDEXES ARE IN THE RIGHT ORDER.  
 BASICALLY, THIS TEST STRUCTURE IS SIMILAR TO ALL PREVIOUS NRA TESTS.  
 IN THE 1ST OF THE 2 N=101., ISOURCE=-1, IDEST=10.  
 IN THE SECOND ONE N=101., ISOURCE=-10., IDEST=1  
 D) THE LAST TEST FOR NRA USES THE SAME DATA AS BEFORE (SEQUENTIAL #'S AT FDATA).  
 BUT N, ISOURCE, IDEST ARE ALL VARIED RANDOMLY. PLEASE REFER TO LISTINGS FOR MORE INFO.  
 7.1-4 THE LDRP AND ARSP ARE CHECKED IN A VERY SIMPLE MANNER. THE FULL LOAD INSTRUCTION IS DONE. THEN THE P-BLOCK IS RELOADED WITH ALL VALUES CHANGING. ONLY CERTAIN VALUES SHOULD EFFECT THE PARTIAL LOAD INSTRUCTION EXECUTION. THIS IS WHAT IS TO BE VERIFIED.  
 FOLLOWING THE CODE SHOULD BE SIMPLE. PLEASE REFER TO AP PROGRAMMERS REFERENCE FOR MORE INFO. ON PARTIAL LOADS.

10008 APJ.T

```

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

```

?WPD 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, HOWEVER 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 INERPRETATION  
 VALUE VALUE  
 1 40000 1 LOOP ON ERROR  
 2 20000 1 SKIP LOOPING ON ERROR  
 3 10000 1 PRINT TO CONSOLE  
 4 04000 1 ABORT PRINT OUT TO CONSOLE  
 5 02000 1 DO NOT PRINT % FAILURE  
 6 01000 1 PRINT % FAILURE  
 7 0 0 ALLOW END OF PASS PRINT OUT  
 8 00400 1 SUPPRESS END OF PASS PRINT OUT  
 9 00200 1 PRINT ON THE LINE PRINTER  
 10 0 0 DO NOT PRINT ON THE LINE PRINTER  
 11 0 0 DO NOT HALT ON ERROR  
 12 0 0 HALT ON ERROR  
 13 0 0 DO NOT PRINT SUMMARY AND/OR  
 14 0 0 PASSING OF EACH SUBTEST  
 15 0 0 PRINT SUMMARY AND/OR  
 16 0 0 PASSING OF EACH SUBTEST  
 17 0 0 PRINT ONLY THE FIRST ERROR  
 18 0 0 PRINT EVERY ERROR  
 SWITCH COMMANDS  
 18.1 ONCE THE PROGRAM STARTS EXECUTING 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 AFFILIATED WITH IT, THUS BIT 4 CAN BE ALTERED BY HITTING KEY 4. (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.  
 18.2

```

0009 APJ.T
01
02
03
04
05
06
07
08
09
10
11
12
13
14
15
16
17
18
19
20
21
22

18.2.1 OTHER COMMANDS
"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 "SWREG"
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 ODT (NOTE: THIS IS AN
OPTIONAL COMMAND AND IS AVAILABLE ONLY IF
OOTPK IS PRESENT)
M THIS COMMAND GIVEN AT ANY TIME WILL PRINT THE
CURRENT OPERATING MODES.

10010 APJ.T
01
02
03
04
05
06
07
08
09
10
11
12
13
14
15
16
17
18
19
20
21

19.0 OPERATING PROCEDURE/OPERATOR INPUT
9.1 OPERATING PROCEDURE
1. LOAD THE PROGRAM IN THE COMPUTER
USING THE BINARY LOADER OR DTOS TAPE.
2. SET SWITCHES TO ONE OF THE FOLLOWING
STARTING ADDRESSES:
500,200 ~
501 ~
502 ~
503 ~
504 ~
505 ~
RUN ALL TESTS
IMR INSTRUCTION TEST (SEQUENTIAL DATA)
IMR INSTRUCTION TEST (RANDOM DATA)
LDC,STC INSTRUCTION TESTS
MRA INSTRUCTION TEST
LDRP,ARSP TESTS

3. PRESS START.
4. THE PROGRAM PRINTS "PASS" FOLLOWING
SUCCESSFUL COMPLETION OF THE TEST.

```



!0013 APJ.T

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

11.3 COMMON SUBROUTINES

A) ADDSET, SETP, LOOP -  
THESE ALL WORK IN CONJUNCTION TO  
SET UP THE TEST LOOP.  
ADDSET TELLS SETP WHERE THE LOOP ENDS.  
SETP MARKS THE START OF THE TEST LOOP.  
LOOP MARKS THE END OF THE TEST LOOP.  
THE AP INSTRUCTION IS EXECUTED AND THE RESULTS ARE  
CHECKED FOR ERROR WITHIN THIS TEST LOOP. IF AN ERROR  
OCCURS, THEN THE PROGRAM CAN LOOP INDEFINITELY BETWEEN  
THE SETP AND THE LOOP. (SEE SECTION 8 ON VARIOUS SWITCH  
SETTINGS).  
NORMALLY, WITH NO ERRORS, THE PROGRAM WILL GO THROUGH  
THIS LOOP ONCE ON THE FIRST PASS OF THE PROGRAM, ON  
SUBSEQUENT PASSES THE PROGRAM WILL STAY IN THE LOOP  
FOR 5 LOOPS. (OR 100, DEPENDING ON THE PROGRAM).  
B) PARAMETER BLOCK LOADER ROUTINES -  
THESE CALLS ALL HAVE DIFFERENT NAMES DEPENDING ON THE  
INSTRUCTION, HOWEVER THEY OPERATE IN THE SAME MANNER.  
THE COMMENT FIELD ON THE CALL WILL ALWAYS INDICATE  
THAT THIS IS A PARAMETER BLOCK LOAD ROUTINE AND WILL  
NAME THE AP INSTRUCTION THAT THE LOAD IS COMPATIBLE WITH.  
THE ARGUMENTS OF THE CALL ARE THE PARAMETERS OF THE AP  
INSTRUCTION. THESE PARAMETERS ARE TO BE LOADED  
BY THESE ROUTINES INTO MAIN MEMORY AT A LOCATION POINTED  
TO BY AC2. AC2 MUST ALWAYS POINT TO THE PARAMETER BLOCK  
DURING AP INSTRUCTION EXECUTION.

!0014 APJ.T

01  
02  
03  
04  
05  
06  
07  
08  
09  
10  
11  
12  
13  
14  
15  
16  
17  
18  
19

C) SLICING ROUTINES -  
THESE TYPES OF ROUTINES EXIST IN SEVERAL DIFFERENT FORMS.  
THEY ALL HAVE VERY SIMILAR NAMES. FOR EXAMPLE - SLICE,  
SLICC, SLICA, SLICB. THESE ARE ALL JSR'S TO SOME SLICING  
ROUTINE.  
THE SLICING ROUTINES WILL SET UP VARIOUS RANDOM  
SLICES IN AP RAM FOR THE VARIOUS INPUT AND OUTPUT  
ARRAYS USED IN THE AP INSTRUCTIONS.  
TO CREATE THESE SLICES, RANDOM N AND STARTING ADDRESS  
ARE CREATED. THE COMMENTS ON THE CALL WILL INDICATE  
THE DESTINATION OF THESE RANDOM PARAMETERS WITHIN THE  
TEST CODE. THE ROUTINES ALL SLICE DIFFERENTLY.  
ONE WILL DIVIDE THE RAM IN HALF AND THEN FIND  
A RANDOM SLICE WITHIN EACH HALF. ANOTHER MIGHT  
FIND A RANDOM SLICE FOR EACH THIRD OF RAM.



10015 APJ.1

11.4 SEQUENCE OF TESTING

IT IS IMPORTANT THAT THE PROGRAMS BE EXECUTED IN A SPECIFIC SEQUENCE:

- APIS DIAG (CPU/AP ONLY)
- APFP DIAG (CPU/AP ONLY)
- APA EXER
- APB EXER
- APC EXER
- APD EXER
- APE EXER
- APF EXER
- APG EXER
- APH EXER
- API EXER
- APJ EXER
- APK EXER

10016 APJ.1

020TD 12

OCTAL DEBUG TOOL (ODT)

THE DIAGNOSTIC IS EQUIPPED WITH A BUILT IN ODT WHICH CAN BE ACCESSED BY HITTING CONTROL 0 (-0) 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:

- ? PRESSED ANY ILLEGAL KEY CAUSES THE ODT TO RESPOND WITH A "2".
- @ 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

FOR ACCUMULATORS 0-3

FOR PC OF THE NEXT INSTRUCTION TO BE EXECUTED IN THE EVENT OF A "P" COMMAND.

CPU AND TIO STATUS

BIT INTERPRETATION

15 STATUS OF TIO 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 AND PRINT ITS CONTENTS.
- .-"ADR"/ SUBTRACT "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.



10019 APJ.T

01 :14.0 RUN TIME

02 :

03 : 14.1 PASS 1

04 :

05 : 14.2 SUBSEQUENT PASSES

06 :

07 :TITL APJ.TX

08 :END

\*\*000000 TOTAL ERRORS, 00000 PASS 1 ERRORS

0020 APJ.T

070TD 001551 MC 16/01

S2WPD 001075 MC 8/01

40 SECONDS

2.5 MINUTES