DataGeneral

TECHNICAL STATEMENT

TEXT LISTING

068-001067-01

PROGRAM

ARRAY PROCESSOR EXERCISER-C

TEXT TAPE

097-001067-01

ABSTRACT

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

```
 0001 APC.T    MACRO REV 06.30          09:37:54 09/19/79
01                      ;*************************************************************
02                      ;
03                      ; NAME: APC.TX                              PART NUMBER 097-1067
04                      ;
05                      ;
06                      ; DESCRIPTION: ARRAY PROCESSOR EXERCISER-C.
07                      ;
08                      ;
09                      ; REVISION HISTORY:
10                      ;
11                      ;       REV.            DATE
12                      ;
13                      ;       00              03/29/78
14                      ;       01              12/29/78
15                      ;
16                      ;
17                      ; COPYRIGHT (C) DATA GENERAL CORPORATION, 1978
18                      ; ALL RIGHTS RESERVED.
19                      ;*************************************************************
```

```
10002 APC.T
01
02                      ;
03                      ;       APC.TX                  PART NUMBER: 097-1067
04                      ;
05                      ;
06                      ;1.0    PROGRAM NAME
07                      ;
08                      ;       SOURCE FILE:    APC.SR
09                      ;       DTOS FILE:      APC EXER
10                      ;
11                      ;2.0    REVISION HISTORY
12                      ;
13                      ;       DATE            REVISION
14                      ;       ----            --------
15                      ;       XX/XX/78           00
16                      ;
17                      ;3.0    MACHINE REQUIREMENTS
18                      ;
19                      ;       1. ECLIPSE FAMILY CENTRAL PROCESSOR  (HOST OR IOP)
20                      ;          WITH AT LEAST 16.K READ/WRITE MEMORY
21                      ;       2. ARRAY PROCESSOR BOARDS AP1, AP2, AP3
22                      ;       3. BASIC I/O TELETYPE INTERFACE AND CONTROL
23                      ;
24                      ;4.0    TEST REQUIREMENTS
25                      ;
26                      ;       1. SAME AS MACHINE REQUIREMENTS
27                      ;       2. I/O TESTER OR DISC
28                      ;       3. MMPU1
29                      ;
30                      ;5.0    SUMMARY
31                      ;
32                      ;       THIS PROGRAM IS A FUNCTIONAL TEST FOR
33                      ;       THE ARRAY PROCESSOR (AP).  IT IS EXECUTED BY
34                      ;       A CENTRAL PROCESSOR (OR IOP) CONTROLLING THE AP
35                      ;       AND TESTS SPECIFIC AP INSTRUCTIONS.
36                      ;
37                      ;6.0    RESTRICTIONS
38                      ;
39                      ;       THE PROGRAM ASSUMES
40                      ;       - THE SYSTEM EXCLUDING THE AP IS ERROR FREE
41                      ;       - THE HOST COMPUTER AND MEMORY/MAP SYSTEMS
42                      ;         ARE WORKING PROPERLY.
43                      ;       - THE 2 AP MAINTAINANCE DIAGNOSTICS USING
44                      ;         THE MAINTAINANCE INSTRUCTION SET (MIS)
45                      ;         HAVE BEEN SUCCESSFULLY RUN.
```

```
10003 APC.T
01                ;7.0    PROGRAM DESCRIPTION/THEORY OF OPERATION
02                ;               APC TEST DESCRIPTION
03                ;
04                ;7.1    APC TESTS THE FOLLOWING INSTRUCTIONS:
05                ;
06                ;       ARA = ADD REAL ARRAYS
07                ;       SRA = SUBTRACT REAL ARRAYS
08                ;       CMA = COMPARE REAL ARRAYS
09                ;       MRA = MULTIPLY REAL ARRAYS
10                ;       MCA = MULTIPLY COMPLEX ARRAYS
11                ;       SPR = SIGNED PRODUCT OF REAL ARRAYS
12                ;       SPS = SIGNED PRODUCT OF SCALAR AND REAL ARRAY
13                ;
14                ;7.1.1  TESTING OF ARA
15                ;       THE MAIN CONCEPT INVOLVED IN MOST OF THIS TEST
16                ;       CAN BE DESCRIBED AS FOLLOWS:
17                ;       GIVEN A SEQUENTIAL, FLOATING POINT ARRAY "X"
18                ;       OF INDEX VALUE "I".
19                ;       AND A SEQUENTIAL, FLOATING POINT ARRAY "Y" OF
20                ;       INDEX VALUE "J" AND STARTING AT 0.0.  THEN
21                ;       (X) + (Y) = (Z)
22                ;               WHERE (X) MEANS ARRAY "X"
23                ;
24                ;       WHERE THE ARRAY "Z" WOULD BE EQUAL TO THE ARRAY
25                ;       "X" ONLY NOW INDEXED BY I + J.
26                ;       THIS CONCEPT ENABLES US TO EXERCISE THE INDEXING
27                ;       FUNCTIONS OF ARA AT THE SAME TIME TESTING VARIATION
28                ;       OF N AND OTHER PARAMETER VARIATIONS.
29                ;
30                ;       A) THE FIRST ARA TEST IS SIMPLE.  THE AP RAM IS
31                ;       DIVIDED INTO THREE SECTIONS.  THE LOWER SECTION IS
32                ;       LOADED WITH SEQUENTIAL FLOATING POINT DATA FROM THE
33                ;       BUFFER IN MAIN MEMORY STARTING AT FDATA.  THE
34                ;       SECOND SECITON IS LOADED WITH 1.0'S.  THESE TWO
35                ;       ARRAYS ARE USED AS INPUT FOR THE ARA INSTRUCTION.
36                ;       THE OUTPUT IS A SEQUENTIAL FLOATING POINT ARRAY.
37                ;       THIS WILL BE STORED IN THE UPPER ONE THIRD OF THE
38                ;       AP RAM.  THIS ARRAY SHOULD COMPARE EQUALLY TO THE
39                ;       ARRAY STARTING AT FDATA+2 IN MAIN MEMORY.  AND
40                ;       INDEED THIS IS CHECKED WITH THE RCOMP ROUTINE.
41                ;       NUMBER OF ELEMENTS IS VARIED FROM 2 TO 1252 BY 2'S.
42                ;       ALL INDEXES ARE EQUAL TO ONE.
43                ;
44                ;       B) THE SECOND TEST IS SIMILAR TO THE FIRST TEST.
45                ;       ONCE AGAIN SEQUENTIAL FLOATING POINT NUMBERS, FROM
46                ;       THE BUFFER STARTING AT FDATA, ARE LOADED TO THE
47                ;       LOWER ONE THIRD OF AP RAM.  TO THE 2ND ONE THIRD
48                ;       OF AP RAM GOES A BUFFER LOADED WITH 1.0'S. THIS
49                ;       BUFFER STARTS AT IDATA IN MAIN MEMORY.  THEN A
50                ;       RANDOM SLICE OF BOTH THESE SECTIONS IN RAM IS TAKEN.
51                ;       THESE SLICES ARE USED AS THE INPUT ARRAYS TO THE
52                ;       ARA INSTRUCTION.  THE OUTPUT ARRAY IS STORED IN
53                ;       ANOTHER RANDOM SLICE IN THE TOP ONE THIRD OF AP
54                ;       RAM.  THIS ARRAY IS COMPARED TO THE CORRESPONDING
55                ;       SLICE OF THE FDATA BUFFER.  THIS TEST CHECKS OUT
56                ;       VARIATION OF INPUT AND OUTPUT STARTING ADDRESSES
57                ;       AND VARIATION OF N.
58                ;
```

```
10004 APC.T
01
02                ;       C) TESTS #3 AND #4 CHECK OUT THE INDEXING CAP-
03                ;       ABILITIES OF THE ARA INSTRUCTION USING THE
04                ;       CONCEPT DISCUSSED AT THE BEGINNING OF THIS SECTION.
05                ;       IN THESE TESTS THE TWO INPUT ARRAYS ARE SEQUENTIAL
06                ;       FLOATING POINT ARRAYS LOADED FROM THE FDATA BUFFER
07                ;       INTO AP RAM.  IN BOTH TESTS (#3 AND #4) THE INPUT
08                ;       INDEXES ARE 6 AND 1.  THEREFORE THE OUTPUT ARRAY
09                ;       SHOULD COMPARE TO A SEQUENTIAL FLOATING POINT ARRAY
10                ;       WITH AN INDEX OF 7.  THIS IS CHECKED.  N IS FIXED
11                ;       IN ALL CASES AT 112.
12                ;
13                ;       D) TEST #5 CHECKS OUT THE OUTPUT INDEXING FUNCTION
14                ;       OF THE ARA INSTRUCTION.  THIS IS DONE WITH INPUTS
15                ;       OF A SEQUENTIAL FLOATING POINT ARRAY AND AN ARRAY
16                ;       OF ALL 1.0'S.
17                ;       THE ARA INSTRUCTION IS THEN EXECUTED WITH AN OUTPUT
18                ;       INDEX OF NON-UNITY.  RESULTS ARE CHECKED BACK WITH
19                ;       FDATA BUFFER.
20                ;       E) TEST #6 AND #7 USE THE INDEXING CONCEPT EXPLAINED
21                ;       IN THE BEGINNING OF THIS SECTION.
22                ;       IN BOTH TESTS N, IJ, IK, IL ARE VARIED.  IN #6
23                ;       VARIATION IS SEQUENTIAL. IN #7 VARIATION IS RANDOM.
24                ;       THE DATA USED IS THE SEQUENTIAL DATA LOADED FROM
25                ;       THE FDATA BUFFER.
26                ;
27                ;7.1.2  THE SRA INSTRUCTION TESTS ARE ALMOST IDENTICAL TO
28                ;       THE ARA TESTS.  EXCEPTIONS ARE NOTED IN THE COMMENTS.
29                ;
30                ;NOTE:  IN BOTH ARA AND SRA TESTS WE HAVE AVOIDED SIMULATION.
31                ;       THIS ENABLES US TO DO A MUCH MORE EXHAUSTIVE TEST.
32                ;
```

```
10005 APC.T
01
02              ;7.2.3  MRA TESTS ARE MORE STRAIGHT FORWARD.  HERE WE SIM-
03              ;       ULATE MRA INSTRUCTION.
04              ;
05              ;       A) TEST #1 VARIES N FROM 2-1252(8).  THE DATA IS
06              ;       SEQUENTIAL FLOATING POINT DATA LOADED FROM FDATA
07              ;       AND FDATA+26.
08              ;       BOTH INPUT ARRAYS ARE LOADED INTO THE LOWER 2/3 OF AP
09              ;       RAM.  THE OUTPUT GOES TO THE TOP 1/3.  THE INDEXES=1.
10              ;
11              ;       THE INSTRUCTION IS SIMULATED AND THE OUTPUT GOES TO
12              ;       THE IDATA BUFFER.  THE INSTRUCTION IS THEN EXECUTED
13              ;       AND SIMULATED AND AP DATA ARE COMPARED.
14              ;
15              ;       B) IN TEST #2 WE SLICE THE TWO INPUT ARRAYS AND
16              ;       THE OUTPUT ARRAY GOES TO A RANDOM SLICE.
17              ;
18              ;NOTE:  IN THIS TEST THE SIMULATION OF MRA INSTRUCTION IS
19              ;       DONE IN THE BEGINNING THE REAL VARIATION IS DONE
20              ;       IN THE SLICING.  THIS WILL CHECK VARIOUS START
21              ;       ADDRESS AND N'S.
22              ;
23              ;       C) TESTS #3 AND #4 VARY N, IJ, IK, IL.  OTHERWISE
24              ;       IT IS SIMILAR TO ALL PREVIOUS TESTS OF MRA.
25              ;       THE INPUT ARRAYS ARE LOADED TO AP RAM FROM SECTIONS
26              ;       OF FDATA.  THE INSTRUCTION IS SIMULATED AND THEN
27              ;       EXECUTED AND RESULTS ARE COMPARED.  SIMULATED
28              ;       RESULTS ARE  ATIDATA.
29              ;
30              ;       TEST #4 IS RANDOM VARIATIONS
31              ;       TEST #3 IS SEQUENTIAL VARIATIONS
32              ;
33              ;7.2.4  SPR TESTS ARE IDENTICAL TO MRA EXCEPT A DIFFERENT
34              ;       SIMULATOR IS USED.
35              ;
```

```
10006 APC.T
01
02              ;7.2.5  SPS TEST DESCRIPTION.
03              ;       THESE TESTS USE THE SEQUENTIAL FLOATING DATA  ATIDATA.
04              ;       AND A SCALAR OF 2.0.  ALL TESTS USE THE SIMULATOR
05              ;       APPROACH.
06              ;       A) TEST #1 VARIES N FROM 0-1024 BY
07              ;
08              ;       B) TEST #2 VARIES N AND IJ RANDOMLY
09              ;
10              ;       C) TEST #3 IS A RANDOM SLICE TEST.
11              ;
12              ;7.2.6  MCA TESTS ARE THE SAME AS MRA TESTS.
13              ;       THE CMA TESTS ARE AN EXHAUSTIVE SET OF TESTS.
14              ;       EACH OPERATOR IS TESTED WITH THREE BASIC TESTS.
15              ;
16              ;       THE DATA USED IS TWO ARRAYS.  ONE AT FDATA AND
17              ;       ONE AT GDATA IN MM.
18              ;       ONE IS A SEQUENTIAL VECTOR AND THE OTHER IS THE
19              ;       INVERSE SEQUENTIAL VECTOR.  THE CMA INSTRUCTION
20              ;       IS SIMULATED IN ALL TESTS.
21              ;       A) VTES1 - THIS STRUCTURE VERIFIES, FOR EACH
22              ;       OPERATOR THAT N CAN BE VARIED, WITH ALL OTHER
23              ;       VARIABLES HELD CONSTANT.  N GOES FROM 2-2000(8)
24              ;       BY 2'S.  SEQUENTIAL DATA FROM FDATA IS LOADED
25              ;       STARTING AT RAM LOC 0.  INVERSE SEQUENTIAL DATA
26              ;       IS LOADED AT RAM LOC 2000.  THE CMA INSTRUCTION
27              ;       IS SIMULATED AND OUTPUT IS STORED IN BDATA ARRAY.
28              ;       THE CMA INSTRUCTION IS DONE AND OUTPUT IS STORED
29              ;       AT IDATA.
30              ;       THE TWO INTEGER VECTORS ARE THEN COMPARED VIA THE
31              ;       ICOMP ROUTINE.
32              ;       IN THIS TESTS N VERIFIES FROM 2-2000 BY 2.
33              ;       B) VTES2 - THIS TEST STRUCTURE WILL TEST VARIOUS
34              ;       INPUT ARRAY STARTING ADDRESS IN RAM AND VARIOUS
35              ;       VALUES OF N.  A RANDOM SLICE IS TAKEN OF EACH
36              ;       INPUT ARRAY.  THE TES1 IS BASICALLY THE SAME AS
37              ;       THE FIRST SET OF TESTS EXCEPT THAT THE INPUT
38              ;       ARRAYS ARE RANDOMLY SLICED.  THE INSTRUCTION IS
39              ;       SIMULATED AND COMPARED TO AP RESULTS.
40              ;       C) THE VTES3 STRUCTURE.
41              ;       THIS STRUCTURE VARIES THE N, IJ, IK FOR ANY CMA
42              ;       OPERATOR.
43              ;       THE SAME DATA IS USED AS IN PREVIOUS CMA TESTS AND
44              ;       IS LOADED INTO AP RAM IN THE SAME MANNER AS IN
45              ;       PREVIOUS TEST.
46              ;       N VARIES FROM 1-5(8)
47              ;       IJ, IK VARY FROM 1-5(8)
48              ;
49              ;       THE INSTRUCTION IS SIMULATED AND THEN EXECUTED.
50              ;       SIMULATE RESULTS ARE STORED IN BDATA ARRAY.
51              ;       AP RESULTS ARE AT IDATA.
52              ;       BDATA IS THEN COMPARED TO IDATA.
53              ;
```

```
!0007 APC.T
01
02              ;NOTE:  THIS TEST IS OF A NESTED DO LOOP TYPE AND :. IS
03              ;       QUITE LONG.
04              ;       D) THE FINAL TEST IS ONE WHICH VARIES N (1-77)
05              ;       IJ, IK VARIES (1-77) AND OPERATOR IS VARIED ALL
06              ;       RANDOMLY.  IN ORDER TO VARY THE OPERATOR OF THIS
07              ;       INSTRUCTION BITS 1-4 OF THE 2ND WD MUST BE
08              ;       VARIED.  IN ORDER TO DO THIS A CMA LT IS AS-
09              ;       SEMBLED IN SCRATCH AREA.  BITS 1-4 ARE VARIED
10              ;       RANDOMLY AND THEN THE ENTIRE INSTRUCTION (BOTH
11              ;       WDS) ARE STORED INTO THE TEST.
12              ;       THIS IS DONE EACH TIME N, IJ, IK ARE CHANGED.
```

```
!0008 APC.T
01                      S?WPD 8
02
03
04              ;8.     SWITCH SETTINGS
05              ;
06              ;       LOCATION "SWREG" IS USED TO SELECT THE PROGRAM OPTIONS
07              ;       (NOT SYSTEM CONFIGURATION). WHILE RUNNING UNDER DTOS,
08              ;       THIS LOCATION WILL BE LOADED BY THE MONITOR.
09              ;       HOWEVER UNDER STAND ALONE AND PROGRAM LOAD MODES THIS
10              ;       LOCATION WILL BE SET ACCORDING TO THE ANSWERS SUPPLIED
11              ;       BY THE OPERATOR. IN ANY CASE THE OPTIONS CAN BE CHANGED
12              ;       OR VERIFIED BY USING ONE OF THE COMMANDS GIVEN IN SEC.
13              ;       8.2
14              ;
15              ;
16              ;8.1    SWITCH OPTIONS
17              ;       DIFFERENT BITS AND THEIR INTERPRETATION AT LOCATION
18              ;       "SWREG" IS AS FOLLOWS:
19              ;
20              ;       BIT     OCTAL   BINARY  INERPRETATION
21              ;               VALUE   VALUE
22              ;
23              ;       1               0       LOOP ON ERROR
24              ;               40000   1       SKIP LOOPING ON ERROR
25              ;
26              ;       2               0       PRINT TO CONSOLE
27              ;               20000   1       ABORT PRINT OUT TO CONSOLE
28              ;
29              ;       3               0       DO NOT PRINT % FAILURE
30              ;               10000   1       PRINT % FAILURE
31              ;
32              ;       4               0       ALLOW END OF PASS PRINT OUT
33              ;               04000   1       SUPPRESS END OF PASS PRINT OUT
34              ;
35              ;       5               0       DO NOT PRINT ON THE LINE PRINTER
36              ;               02000   1       PRINT ON THE LINE PRINTER
37              ;
38              ;       6               0       DO NOT HALT ON ERROR
39              ;               01000   1       HALT ON ERROR
40              ;
41              ;       7               0       DO NOT PRINT SUMMARY AND/OR
42              ;                               PASSING OF EACH SUBTEST
43              ;               00400   1       PRINT SUMMARY AND/OR
44              ;                               PASSING OF EACH SUBTEST
45              ;       8               0       PRINT ONLY THE FIRST ERROR
46              ;               00200   1       PRINT EVERY ERROR
47
48              ;8.2    SWITCH COMMANDS
49              ;       ONCE THE PROGRAM STARTS EXECUTING THE STATE OF ANY OF
50              ;       THE BITS CAN BE CHANGED BY HITTING KEYS 1-9, A-F. THE
51              ;       PROGRAM WILL CONTINUE RUNNING AFTER UPDATING THE OPTIONS.
52              ;       EACH KEY WILL COMPLEMENT THE STATE OF THE BIT AFFILIAT-
53              ;       ED WITH IT, THUS BIT 4 CAN BE ALTERED BY HITTING KEY 4.
54              ;       SETTING OF ANY BIT OF LOCATION "SWREG" WILL SET BIT 0.
55              ;       (DEFAULT MODE IS DEFINED AS ALL BITS OF SWREG SET TO 0)
56              ;       THE PROGRAM CAN BE LOCKED INTO SWITCH MODIFICATION MODE
57              ;       BY TYPING A 0, IN WHICH CASE MORE THAN ONE BIT CAN BE
58              ;       CHANGED BEFORE  CONTROL IS ALLOWED TO RETURN TO THE
59              ;       MAIN PROGRAM.
60              ;
```

```
 0009 APC.T
01              ;8.2.1  OTHER COMMANDS
02              ;
03              ;       "CR"    A "RETURN" CAN BE TYPED TO CONTINUE THE PROGRAM
04              ;               AFTER ITS LOCKED IN A SWITCH MODIFICATION MODE
05              ;
06              ;       ^D      THIS COMMAND GIVEN AT ANY TIME WILL RESET "SWREG"
07              ;               TO DEFAULT MODE AND RESTART THE PROGRAM.
08              ;
09              ;       ^R      THIS COMMAND GIVEN AT ANY TIME WILL RESTART THE
10              ;               PROGRAM. SWITCHES ARE LEFT WITH THE VALUES THEY
11              ;               HAD BEFORE THE COMMAND WAS ISSUED.
12              ;
13              ;       ^O      THIS COMMAND GIVEN AT ANY TIME WILL CAUSE THE
14              ;               PROGRAM CONTROL TO GO TO ODT (NOTE: THIS IS AN
15              ;               OPTIONAL COMMAND AND IS AVAILBLE ONLY IF
16              ;               ODTPK IS PRESENT)
17              ;
18              ;       M       THIS COMMAND GIVEN AT ANY TIME WILL PRINT THE
19              ;               CURRENT OPERATING MODES.
20              ;
21
22              ;
```

```
10010 APC.T
01              ;9.0    OPERATING PROCEDURE/OPERATOR INPUT
02              ;
03              ;       9.1     OPERATING PROCEDURE
04              ;               1. LOAD THE PROGRAM IN THE COMPUTER
05              ;                  USING THE BINARY LOADER OR DTOS TAPE.
06              ;               2. SET SWITCHES TO TO ONE OF THE FOLLOWING
07              ;                  STARTING ADDRESSES:
08              ;                     500,200 -     RUN ALL TESTS
09              ;                     501 -         ARA INSTRUCTION TEST
10              ;                     502 -         SRA INSTRUCTION TEST
11              ;                     503 -         MRA INSTRUCTION TEST
12              ;                     504 -         SPR INSTRUCTION TEST
13              ;                     505 -         SPS INSRTUCTION TEST
14              ;                     506 -         MCA INSTRUCTION TEST
15              ;                     507 -         CMA INSTRUCTION TEST
16
17              ;
18              ;
19              ;
20              ;
21              ;
22              ;               3. PRESS START.
23              ;               4. THE PROGRAM PRINTS "PASS" FOLLOWING
24              ;                   EACH SUCCESSFUL COMPLETION OF THE TEST
25              ;
```

```
10011 APC.T
01              ;10.   ERRORS
02              ;       WHEN AN ERROR IS DETECTED A VALUE FOR AN
03              ;       ERROR PC WILL BE PRINTED ALONG WITH AC'S 0,1 AND 2
04              ;       OF THE HOST COMPUTER.
05              ;
06              ;       THE PC VALUE PRINTED MAY POINT TO ONE OF THE
07              ;       FOLLOWING:
08              ;
09              ;               1. A CALL TO EHALT
10              ;               2. A CALL TO THE ROUTINE WHICH DETECTED
11              ;                  THE ERROR AND SUBSEQUENTLY EXITED
12              ;                  TO EHALT SUCH AS THE VECTOR COMPARE
13              ;                  ROUTINES TCOMP,CCOMP,RCOMP, AND RCCMP.
14              ;
15              ;       SUBROUTINES OF THIS TYPE HAVE SPECIAL ERROR PRINTOUTS
16              ;       CALLED IN THE ERROR ROUTINE. IN THE CASE OF THE REAL
17              ;       VECTOR COMPARE ROUTINES, TCOMP AND RCOMP, UP TO 5
18              ;       PAIRS OF GOOD/BAD DATA IS PRINTED IN 2-16. BIT HEX
19              ;       OR OCTAL FORMAT (SEE SW10), ALONG WITH THE MM ADDRESS
20              ;       OF THE DATA AND THE RELATIVE INDEX WITHIN THE VECTOR.
21              ;       A TOTAL ERROR COUNT IS PRINTED ALONG WITH
22              ;       THE VECTOR/VECTOR N,IJ,IK DATA. IN THE CASE OF THE
23              ;       COMPLEX VECTOR COMPARE ROUTINES, CCOMP AND RCCMP, THE
24              ;       GOOD/BAD DATA IS PRINTED IN 4-16. BIT HEX OR OCTAL
25              ;       FORMAT.
26              ;
27              ;       THE SIGNIFICANCE OF ERROR PRINTOUTS IS EXPLAINED IN THE
28              ;       LISTING AT THE SOURCE OF THE ERROR.  IN GENERAL
29              ;       AC0 CONTAINS THE CHECKWORD AND AC1 THE DATA IN
30              ;       QUESTION.
31              ;
32              ;       AFTER PRINTING THE ABOVE THE PROGRAM HALTS ALLOWING
33              ;       THE OPERATOR TO SET THE SWITCHES.  UPON PRESSING
34              ;       CONTINUE THE PROGRAM MAY EITHER GO ON TO THE NEXT
35              ;       TEST OR GO INTO A SCOPE LOOP BETWEEN THE APPROPRIATE
36              ;       CALLS TO .SETUP AND .LOOP (SEE 8.0 SWITCH SETTINGS)
37
```

```
10012 APC.T
01
02              ;11.0   DEBUG HELP
03              ;
04              ;       11.1    MONITOR LOCATIONS
05              ;
06              ;               LOCATION        LABEL   MEANING
07              ;                 200                   DTOS START
08              ;                 201                   FIRST LOCATION OF CURRENT TEST
09              ;                 202                   START ADDRESS
10              ;                 203                   XXXX
11              ;                 204                   PASS COUNT
12              ;
13              ;       11.2    PROGRAMMING DESCRIPTION
14              ;
15              ;                SEE PROGRAMMERS REFERENCE
16              ;               FOR ARRAY PROCESSOR.
17              ;
```

```
10013 APC.T
01
02                ;       11.3    COMMON SUBROUTINES
03                ;
04                ;
05                ;       A) ADDSET,SETP,LOOP -
06                ;       THESE ALL WORK IN CONJUNCTION TO
07                ;       SET UP THE TEST LOOP.
08                ;       ADDSET TELLS SETP WHERE THE LOOP ENDS.
09                ;       SETP MARKS THE START OF THE TEST LOOP.
10                ;       LOOP MARKS THE END OF THE TEST LOOP
11                ;       THE AP INSTRUCTION IS EXECUTED AND THE RESULTS ARE
12                ;       CHECKED FOR ERROR WITHIN THIS TEST LOOP. IF AN ERROR
13                ;       OCCURS, THEN THE PROGRAM CAN LOOP INDEFINITELY BETWEEN
14                ;       THE SETP AND THE LOOP.(SEE SECTION 8 ON VARIOUS SWITCH
15                ;       SETTINGS).
16                ;
17                ;       NORMALLY, WITH NO ERRORS, THE PROGRAM WILL GO THROUGH
18                ;       THIS LOOP ONCE ON THE FIRST PASS OF THE PROGRAM. ON
19                ;       SUBSEQUENT PASSES THE PROGRAM WILL STAY IN THE LOOP
20                ;       FOR 5 LOOPS.(OR 100. DEPENDING ON THE PROGRAM).
21                ;
22                ;
23                ;       B) PARAMETER BLOCK LOADER ROUTINES -
24                ;       THESE CALLS ALL HAVE DIFFERENT NAMES DEPENDING ON THE
25                ;       INSTRUCTION. HOWEVER THEY OPERATE IN THE SAME MANNER.
26                ;       THE COMMENT FIELD ON THE CALL WILL ALWAYS INDICATE
27                ;       THAT THIS IS A PARAMETER BLOCK LOAD ROUTINE AND WILL
28                ;       NAME THE AP INSTRUCTION THAT THE LOAD IS COMPATIBLE WITH.
29                ;       THE ARGUMENTS OF THE CALL ARE THE PARAMETERS OF THE AP
30                ;       INSTRUCTION. THESE PARAMETERS ARE TO BE LOADED
31                ;       BY THESE ROUTINES INTO MAIN MEMEORY AT A LOCATION POINTED
32                ;       TO BY AC2. AC2 MUST ALWAYS POINT TO THE PARAMETER BLOCK
33                ;       DURING AP INSTRUCTION EXECTION.
34
```

```
10014 APC.T
01
02                ;       C) SLICING ROUTINES -
03                ;       THESE TYPES OF ROUTINES EXIST IN SEVERAL DIFFERENT FORMS.
04                ;       THEY ALL HAVE VERY SIMILAR NAMES.FOR EXAMPLE - SLICE,
05                ;       SLICC,SLICA,SLICB.THESE ARE ALL JSR'S TO SOME SLICING
06                ;       ROUTINE.
07                ;       THE SLICING ROUTINES WILL SET UP VARIOUS RANDOM
08                ;       SLICES IN AP RAM FOR THE VARIOUS INPUT AND OUTPUT
09                ;       ARRAYS USED IN THE AP INSTRUCTIONS.
10                ;       TO CREATE THESE SLICES, RANDOM N AND STARTING ADDRESS
11                ;       ARE CREATED. THE COMMENTS ON THE CALL WILL INDICATE
12                ;       THE DESTINATION OF THESE RANDOM PARAMETERS WITHIN THE
13                ;       TEST CODE. THE ROUTINES ALL SLICE DIFFERENTLY.
14                ;       ONE WILL DIVIDE THE RAM IN HALF AND THEN FIND
15                ;       A RANDOM SLICE WITHIN EACH HALF. ANOTHER MIGHT
16                ;       FIND A RANDOM SLICE FOR EACH THIRD OF RAM.
17                ;
18                ;
19                ;
20                ;
```

```
10015 APC.T
01                ;       11.4    SEQUENCE OF TESTING
02                ;
03                ;               IT IS IMPORTANT THAT THE PROGRAMS BE
04                ;               EXECUTED IN A SPECIFIC SEQUENCE:
05                ;
06                ;
07                ;               APIS DIAG (CPU/AP ONLY)
08                ;               APFP DIAG (CPU/AP ONLY)
09                ;               APA EXER
10                ;               APB EXER
11                ;               APC EXER
12                ;               APD EXER
13                ;               APE EXER
14                ;               APF EXER
15                ;               APG EXER
16                ;               APH EXER
17                ;               API EXER
18                ;               APJ EXER
19                ;               APK EXER
```

```
10016 APC.T
01
02                      O?DTD 12
03
04                ;12.  OCTAL DEBUG TOOL (ODT)
05                ;
06                ;       THE DIAGNOSTIC IS EQUIPED WITH A BUILT IN ODT WHICH CAN
07                ;       BE ACCESSED BY HITTING CONTROL O (^O) AT ANY TIME DURING
08                ;       THE EXECUTION OF THE PROGRAM (AFTER SETTING THE PARA-
09                ;       METERS).
10                ;       ON ENTERING ODT THE ADDRESS OF THE LOCATION HAVING THE
11                ;       NEXT INSTRUCTION TO BE EXECUTED WILL BE TYPED-OUT.
12                ;
13                ;12.1  CONVENTIONS AND SYMBOLS
14                ;       THE FOLLOWING CONVENTIONS ARE USED BY THE ODT:
15                ;       ?       PRESSING ANY ILLEGAL KEY CAUSES THE ODT TO RES-
16                ;               POND WITH A "?".
17                ;       @       ODT IS READY AND AT YOUR SERVICE.
18                ;
19                ;12.2  COMMAND STRUCTURE
20                ;       AN ODT COMMAND HAS THE FOLLOWING FORMAT:
21                ;               [ARGUMENT][COMMAND]
22                ;       AN ARGUMENT MAY BE ONE OF THE FOLLOWING:
23                ;       "EXP"   AN OCTAL EXPRESSION CONSISTING OF OCTAL NUMBERS
24                ;               SEPARATED BY PLUS (+) OR MINUS (-) SIGNS. LEAD-
25                ;               ING ZEROS NEED NOT BE TYPED.
26                ;       "ADR"   AN ADDRESS IS THE SAME AS AN EXPRESSION EXCEPT
27                ;               THAT BIT 0 IS NEGLECTED.
28                ;       A COMMAND IS A SINGLE TELETYPE CHARACTER
29                ;
30                ;12.3  ODT COMMANDS
31                ;       THE LOCATIONS THAT CAN BE EXAMINED AND MODIFIED BY THE
32                ;       USER ARE CALLED CELLS. THESE CELLS ARE OF TWO TYPES:
33                ;       INTERNAL CPU CELLS AND MEMORY LOCATIONS.
34                ;
35                ;12.3.1 OPENING INTERNAL CELLS
36                ;       THE COMMAND TO OPEN ONE OF THE INTERNAL REGISTERS IS OF
37                ;       THE FORM "NA" WHERE N IS ANY OCTAL EXPRESSION BETWEEN
38                ;       0 AND 7
39                ;       0-3     FOR ACCUMULATORS 0-3
40                ;       4       FOR PC OF THE NEXT INSTRUCTION TO BE EXECUTED IN
41                ;               THE EVENT OF A "P" COMMAND.
42                ;       5       CPU AND TTO STATUS
43                ;               BIT     INTERPRETATION
44                ;               15      STATUS OF TTO DONE FLAG
45                ;               14      STATUS OF INTERRUPTS (ION FLAG)
46                ;               13      STATUS OF CARRY BIT
47                ;       6       ADDRESS OF THE LOCATION HAVING THE BREAK POINT (IF
48                ;               ANY)
49                ;       7       INSTRUCTION AT THE BREAK POINT LOCATION
50
51                ;       OTHER COMMANDS TO OPEN CELLS ARE:
52                ;
53                ;       "ADR"/  OPEN THE CELL AND PRINT ITS CONTENTS
54                ;       ./      OPEN THE CELL CURRENTLY POINTED TO BY THE POINTER
55                ;               AND PRINT ITS CONTENTS.
56                ;       .+"ADR"/        ADD "ADR" TO THE POINTER, OPEN THE CELL
57                ;               AND PRINT ITS CONTENTS.
58                ;       .-"ADR"/        SUBTRACT "ADR" FROM THE POINTER, OPEN
59                ;               THE CELL AND PRINT ITS CONTENTS.
60                ;       "CR"    THE RETURN KEY IS USED TO CLOSE THE OPEN CELL
```

```
01              ;               WITH OR WITHOUT MODIFICATION.
02              ;       "LF"    LINE FEED IS USED TO CLOSE THE OPEN CELL WITH OR
03              ;               WITHOUT MODIFICATION AND TO OPEN THE SUCCEEDING
04              ;               CELL.
05              ;       ^       CLOSE THE OPEN CELL WITH OR WITHOUT MODIFICATION
06              ;               AND OPEN THE PRECEDING CELL
07              ;       /       CLOSE THE OPEN CELL WITHOUT MODIFICATION, AND
08              ;               OPEN THE CELL POINTED TO BY ITS CONTENTS.
09              ;       +"ADR"/ CLOSE THE OPEN CELL WITHOUT MODIFICATION, AND
10              ;               OPEN THE CELL POINTED TO BY ITS CONTENTS + "ADDR".
11              ;       -"ADR"/ CLOSE THE OPEN CELL WITHOUT MODIFICATION, AND
12              ;               OPEN THE CELL POINTED TO BY ITS CONTENTS - "ADR".
13              ;
14              ;12.3.2 MODIFICATION OF A CELL
15              ;       ONCE A CELL HAS BEEN OPENED ITS CONTENTS CAN BE MODIFIED
16              ;       BY TYPING THE NEW VALUE THE CELL IS TO CONTAIN IN THE
17              ;       FORM OF AN OCTAL EXPRESSION FOLLOWED BY "CR" OR "LF".
18              ;       IF A + OR - IS TYPED AS THE FIRST CHARACTER OF THE EX-
19              ;       PRESSION THEN THE VALUE OF THE EXPRESSION IS ADDED TO OR
20              ;       SUBTRACTED FROM THE OLD CONTENTS OF THE CELL. THE
21              ;       ADDRESS ITSELF OR AN EXPRESION RELATIVE TO THE ADDRESS
22              ;       CAN BE DEPOSITED BY TYPING A "." OR ".+/-OCTAL EXPRESS-
23              ;       ION". A RUBOUT COMMAND GIVEN RIGHT AFTER OPENING A CELL
24              ;       ALLOWS THE MODIFICATION OF ITS CONTENTS AS IF THEY WERE
25              ;       TYPED IN JUST BEFORE THE COMMAND WAS ISSUED.
26              ;
27              ;12.3.3 OTHER ODT COMMANDS
28              ;       RUBOUT  THIS KEY IS USED TO DELETE ERRONEOUSLY TYPED
29              ;               DIGITS. EACH TIME THE KEY IS PRESSED THE RIGHT MOST
30              ;               DIGIT IS DELETED AND ECHOED ON THE TERMINAL. IF
31              ;               THE RUBOUT KEY IS PRESSED RIGHT AFTER OPENING A
32              ;               CELL THEN IT DELETES THE RIGHT MOST DIGIT OF THE CELLS
33              ;               CONTENTS. THIS ALLOWS THE MODIFICATION OF THE CELL
34              ;               AS IF ITS CONTENTS WERE TYPED IN JUST BEFORE THE
35              ;               KEY WAS PRESSED.
36              ;       "ADR"B  INSERT A BREAK POINT AT LOCATION "ADR".
37              ;               ONLY ONE BREAK POINT CAN BE INSERTED AND ANY
38              ;               ENTRY TO ODT AFTER EXECUTING A BREAK POINT WILL
39              ;               CAUSE IT TO BE DELETED.
40              ;       D       DELETE THE BREAK POINT IF ANY.
41              ;       P       RESTART THE EXECUTION OF THE PROGRAM AT LOCATION
42              ;               POINTED BY 4A.
43              ;       "ADR"R  START EXECUTING THE PROGRAM AT "ADR" AFTER AN
44              ;               IO-RESET.
45              ;       K       KILL THE STRING TYPED SO FAR. THE ODT RESPONDS
46              ;               WITH A "?" AND THE OPEN CELL IS CLOSED WITHOUT
47              ;               MODIFICATION.
48              ;       =       PRINT THE OCTAL VALUE OF THE INPUT ONLY.
49              ;               THIS WILL CLOSE ANY OPEN CELLS WITHOUT
50              ;               MODIFICATION AND WILL NOT OPEN A CELL
51              ;
52              ;
53              ;       NOTE:   IN PROGRAMS WHICH RELOCATE THEMSELVES THE
54              ;               THE USER SHOULD PLACE BREAK POINTS ONLY IN THE
55              ;               THE ORIGINAL PROGRAM AREA. IF A BREAK POINT IS
56              ;               PLACED OUTSIDE THIS AREA THE RESULTS WILL
57              ;               BE UNPREDICTABLE.
58              ;
59              ;
```



```
01              ;13.    SPECIAL NOTES/SPECIAL FEATURES
02              ;
03              ;       13.1    FOR A COMPLETE TEST ALL PROGRAMS
04              ;               SHOULD BE EXECUTED WITH CAT/KITTEN.
05              ;
06              ;       13.2    A NOTE ABOUT AP ADDRESSING
07              ;
08              ;               ADDRESSES IN THE AP CAN BE OF SEVERAL
09              ;               MODES:
10              ;
11              ;               1) ONE WORD MODE (1E) SAME AS STANDARD
12              ;                  ADDRESSING
13              ;               2) TWO WORD MODE (2WM) - EACH 32 BITS IS
14              ;                  NOW ONE ADDRESS SPACE.  THIS IS USED
15              ;                  IN THE AP TO SIMPLIFY REAL NUMBER
16              ;                  ADDRESSING.
17              ;               3) FOUR WORD MODE (4WM) - EACH 64 BITS IS
18              ;                  NOW ONE ADDRESS SPACE.  THIS IS USED
19              ;                  IN THE AP TO SIMPLIFY COMPLEX NUMBER
20              ;                  ADDRESSING.
21              ;
22              ;               THE AP ACCESSES AN ADDRESS RELATIVE TO THE START OF
23              ;               THE AP RAM.  THUS AP RAM LOC 0 WOULD BE THE
24              ;               FIRST LOCATION THAT IS IN THE AP.  HOWEVER,
25              ;               AS FAR AS THE ECLIPSE CPU IS CONCERNED, AP
26              ;               LOC 0 IS CONTAINED AT LOCATION LABEL
27              ;               "RAMPT". (IN PAGE ZERO.  SO, IF RAMPT CON-
28              ;               TAINS 64000, THEN 2000 2WM (AP RAM) IS REALLY
29              ;               64000+2000+2000=70000.
30              ;
31              ;               NOTE: "STOP ON STORE" OR "STOP ON ADDRESS" IN AP RAM
32              ;               SPACE WILL NOT WORK IF THE AP IS USING
33              ;               THE INTERNAL AP ADDRESS LINES TO ACCESS AP RAM.
```

```
!0019 APC.T                                                                    0020 APC.T
01                ;14.0   RUN TIME
02                ;
03                ;       14.1    PASS 1                  40 SECONDS
04                ;
05                ;       14.2    SUBSEQUENT PASSES       1 MIN. 40 SECONDS
06                ;
                          .TITL APC.TX
08                        .END

**00000 TOTAL ERRORS, 00000 PASS 1 ERRORS
```

```
O?DTD 001551  MC    16/02
S?WPD 001075  MC     8/01
```