```
10003  N3MPL
01                ;       NOVA3/4 MULTIPROGRAMMING RELIABLITY TEST
02                ;
03                ;1.     ABSTRACT
04                ;       THE NOVA3/4 MULTIPROGRAMMING RELIABILITY TEST
05                ;       CONSISTS OF A SERIES OF INDIVIDUAL PROCESSOR
06                ;       AND PERIPHERAL TESTS AND A
07                ;       SUPERVISOR PROGRAM. (THE DIAGNOSTIC LINKER)
08                ;
09                ;       THE DIAGNOSTIC LINKER IS A PROGRAM
10                ;       DESIGNED TO "LINK" THE VARIETY OF
11                ;       PROCESSOR AND PERIPHERAL TESTS IN
12                ;       SUCH A FASHION THAT THEY MAY BE
13                ;       RUN CONCURRENTLY.  THEREBY, TESTING
14                ;       THE INTERACTIVE CAPABILITIES OF
15                ;       THE PROCESSOR AND ITS PERIPHERAL
16                ;       EQUIPMENT.
17                ;       THIS TEST IS PROVIDED IN THREE LENGTHS
18                ;
19                ;       THE SHORT VERSION ONLY INCLUDES THOSE TESTS
20                ;       THAT APPLY TO THE CPU, MEMORY, FLOATING POINT,
21                ;       MUL\DIV,TTY, REAL TIME CLOCK, AND THE I/O TESTER.
22                ;
23                ;       THE LONG VERSION INCLUDES THE ABOVE + PRIMARY DEVICE
24                ;        CODE TESTS FOR THE NOVA DISK, MOVING HEAD
25                ;       DISK, 6060 SERIES DISK,6063/64 DISK,MAGNETIC TAPE,CASSETTE,
26                ;       DCU-50/200 ,AND THE LINE PRINTER
27                ;
28                ;       THE PERIPHERAL VERSION INCLUDES ONLY THE
29                ;       CHECKERBOARD TEST AS A BACKGROUND TEST BUT
30                ;       DOES INCLUDE THE PRIMARY AND SECONDARY (WITH DCU-50/200 AND
31                ;       LPT AS AN EXCEPTION) DEVICE CODE TESTS FOR
32                ;       THE  DEVICES MENTIONED ABOVE.
33                ;
34                ;2.0    HARDWARE CONFIGURATIONS
35                ;2.1    MINIMUM MACHINE REQUIREMENTS
36                ;2.1.1  NOVA3 OR NOVA4 PROCESSOR
37                ;2.1.2  16K OF READ WRITE MEMORY
38                ;       ( MEMORY MUST BE CONTIGUOUS)
39                ;2.1.3  TTY/CONSOLE (DEV.10/11)
40                ;
41                ;2.2      OPTIONAL EQUIPMENT
42                ;
43                ;2.2.1  UP TO 128K OF READ/WRITE MEMORY
44                ;       (MUST BE CONTIGUOUS)
45                ;2.2.2  MAP OPTION(WITH OR WITHOUT PROTECTION)
46                ;2.2.3  PARITY OPTION
47                ;2.2.4  FLOATING POINT UNIT
48                ;2.2.5  DCU-50/200 FEATURE
49                ;2.2.6  REAL TIME CLOCK (DEV.#14)
50                ;2.2.7  FIXED HEAD DISK (NOVA DISK)(DEV.#20,60)
51                ;2.2.8  MOVING HEAD DISK (ANY/ALL DRIVES)(DEV.#33,73)
52                ;2.2.9  6060 SERIES DISK (ANY/ALL DRIVES)(DEV.#27,67)
53                ;2.2.10 6063/64 DISK (ANY/ALL DRIVES)(DEV.#26,66)
54                ;2.2.11 MAGNETIC TAPE (ANY/ALL DRIVES)(DEV.#22,62)
55                ;2.2.12 CASSETTE (ANY/ALL DRIVES)(DEV.#34,74)
56                ;2.2.13 LINE PRINTER (REG AND DCH TYPE)(DEV.#17)
57                ;2.2.14 I/O TESTER (DEV.#0)
```

```
10004  N3MPL
01                ;
02                ;2.3    PREREQUISITES
03                ;2.3.1  SOFTWARE PREREQUISITES
04                ;       THE SYSTEM SHOULD BE CAPABLE
05                ;       OF RUNNING ALL INDIVIDUAL LOGIC AND
06                ;       RELIABILITY TESTS PERTAINING TO THE
07                ;       PROCESSOR AND ITS PERIPHERAL EQUIPMENT
08                ;       BEFORE ATTEMPTING TO RUN THIS TEST
09                ;NOTE:  ALTHOUGH THIS TEST MAY AT TIMES BE USEFUL
10                ;IN DETERMING THE GO/NO GO STATUS OF AN
11                ;UNKNOWN SYSTEM, IT IS RECOMMENDED THAT:
12                ;A. ALL OTHER DIAGNOSTICS BE RUN EVEN IN THE
13                ;   EVENT THAT THIS TEST FINDS NO PROBLEMS.
14                ;B. AN ATTEMPT BE MADE TO ISOLATE ANY PROBLEMS
15                ;   FOUND BY FIRST UTILIZING THE LOWER
16                ;   LEVEL TESTS FOR MORE CONSISE ERROR REPORTS.
```

```
                ;2.4.2  SYSTEM SETUP
                ;       IF THE DCU IS TO BE RUN AT A DEVICE
                ;       CODE OTHER THAN 64, THEN ONE MEMORY
                ;       LOCATION NEEDS TO BE UPDATED TO
                ;       CONTAIN THE NEW DEVICE CODE.
                ;       THE LOCATION IS:
      015306            LOC.=DCUDV
                ;       IF THE MOVING HEAD DISKS ARE TO BE
                ;       EXERCISED THEY MUST HAVE A PACK INSTALLED
                ;       AND BE IN THE READY STATE
                ;       IF MAGNETIC TAPES ARE TO BE EXERCISED
                ;       THEY MUST BE ON LINE WRITE ENABLED
                ;       IF CASSETTES ARE TO BE EXERCISED
                ;       THEY MUST BE ON LINE WRITE ENABLED
                ;       IF THE LINE PRINTER IS TO BE EXERCISED
                ;       IT MUST BE ON LINE AND IN THE READY STATE
                ;       IF PARITY OPTION EXISTS IT MUST BE
                ;       JUMPED TO INTERRUPT, AND NOT HALT.
                ;
                ;3.     KEY ENTERED OPTIONS
                ;              ENTRIES TYPED ON TTY SET BITS IN SWREG
                ;              FOR USE BY THE PROGRAM.
                ;
                ;KEY    SWREG   BIT     FUNCTION
                ; 1     1       =1 DON'T RELEASE AND ALLOW REASSIGNMENT
                ;                  OF MEMORY AFTER ERROR
                ; 2     2       =1 DELETE TTO OUTPUT
                ; 3     3       =1 CAUSES THE DELETION OF THE RANDOM
                ;                  WAIT STATES IN THE TTY AND LPT
                ;                  TESTS.
                ; 4     4          WILL CAUSE THE ELAPSED RUN
                ;                  TIME AND ACCUMULATED ERRORS
                ;                  TO BE TYPED ON THE TTY.
                ;                  (NOTE: A RTC MUST EXIST)
                ; 5     5       =1 DIRECT ALL ERROR AND RUNTIME TYPEOUTS
                ;                  ALSO TO THE LINE PRINTER.
                ; 6     6       =1 THE ERROR ROUTINE WILL PAUSE AFTER
                ;                  EACH PHASE OF AN ERROR TYPEOUT.
                ;                  TYPE A CR KEY ON DEVICE TTI TO PROCEED.
                ; 7     7       =1 PRINT THE RUN STATISTICS OF EACH TEST.
                ;
                ;       EACH KEY ENTRY COMPLEMENTS THE PREVIOUS STATE OF
                ;       SWREG BIT EXCEPT CONTROL CHARACTERS
                ;       FOLLOWING:
                ;
                ;       KEY (C)O        ENTER THE ODT EDITOR
                ;                       (SEE DESCRIPTION AT PARAGRAPH 7.0)
                ;       KEY (C)D        DEFAULT MODE RESTART. SWREG
                ;                       SET TO 0.
                ;       KEY (C)R        RESTART WITHOUT RESETTING SWREG BITS.
                ;       KEY M           TYPE THE CURRENT CONTENTS OF SWREG.
                ;
                ;       WHERE (C) SIGNIFIES A CONTROL KEY.
```



```
                ;4.      OPERATING PROCEDURES
                ;4.1     LOAD THE PROGRAM VIA THE BINARY LOADER
                ;4.2     SET SWITCHES TO:
                ;       200 FOR AUTO SIZE AND GO
                ;       202 FOR MANUAL SELECT/DELETE
                ;       204 FOR RUNNING WITHOUT MAPPING.
                ;       206 FOR RESTARTING LAST PROGRAM SELECTED
                ;       210 FOR STARTING ODT BEFORE PROGRAM START
                ;
                ;       ****NOTE:  TEST MUST BE RUN BEFORE ADDRESS
                ;                  206 CAN BE UTILIZED.
                ;
                ;4.3     PRESS START
                ;4.4     PROCESSOR WILL TYPE:
                ;       NAME/VERSION
                ;       TOTAL #1K'S=XXX(DECIMAL) MAP OR NO MAP
                ;       PROGRAM RUN LIST
                ;       PROG# DESCRIPTION
                ;4.5     IF START WAS 200 OR 206   THE LIST OF
                ;       PROGRAMS TO BE RUN CONCURRENTLY WILL
                ;       THEN BE LISTED AND THE TEST SYSTEM
                ;       WILL AUTO START
                ;4.6     IF START WAS 202 LINKER WILL
                ;       PAUSE AT THE END OF EACH TEST
                ;       DESCRIPTION AND WAIT FOR KEYBOARD
                ;       INPUT.  TYPING IN A SPACE WILL
                ;       ENABLE THAT TEST TO BE RUN.
                ;       TYPING IN ANY OTHER CHARACTER WILL
                ;       DELETE THAT TEST FROM BEING RUN
                ;
                ;4.7     IF START WAS 204 LINKER WILL SIZE MEMORY
                ;       WITHOUT UTILIZING OR EVEN LOOKING FOR THE
                ;       MAP AND THEN PROCEED AS IN STARTING AT
                ;       ADDRESS 202 WITH THE MAP NON-EXISTENT.
                ;4.8     IF AN AUTOSTART ADDRESS WASN'T USED
                ;       THE PROGRAM WILL WAIT FOR OPTION SETUP
                ;       OF SWREG.  SEE PARAGRAPH 3.0.  TYPE
                ;       A CR TO START TESTING.
```

```
01              ;
02              ;5.     ERROR DESCRIPTION
03              ;       MOST ERRORS DETECTED BY EITHER
04              ;       THE INDIVIDUAL TEST PROGRAMS OR
05              ;       BY THE DIAGNOSTIC LINKER WILL
06              ;       RESULT IN AN EXTENSIVE ERROR
07              ;       TYPEOUT.  SOME SMALL NUMBER OF
08              ;       HIGHLY IMPROBABLE ERRORS MAY RESULT
09              ;       IN A PROGRAM HALT IF THEY ARE
10              ;       OF A NATURE THAT THE LINKER CANNOT
11              ;       RECOVER FROM AND LOGICALLY PROCEED,
12              ;       (I.E. INTERRUPT STACK OVERFLOWS)
13              ;
14              ;
15              ;5.1    ERROR FORMAT
16              ;       ERROR TYPEOUTS INCLUDE
17              ;5.1.1          PROGRAM # AND NAME AT TIME OF ERROR
18              ;       (SEE PROGRAM RUN LIST TO CORRELATE)
19              ;5.1.2          THE CURRENT CONTENTS OF AC0, AC1, AC2.
20              ;5.1.3          LOGICAL SCRATCH AND DATA CHANNEL LIMITS
21              ;5.1.4          MEMORY ALLOCATION TABLE
22              ;       PHYSICAL 1K PAGE# +  LOGICAL ADDRESS +RELOCATED ADDRESS
23              ;5.1.5          CONTINUATION INFORMATION IN GROUPS
24              ;       OF 3 MEMORY LOCATIONS PERTINENT TO
25              ;       THE INDIVIDUAL TEST THAT FAILED
26              ;
27              ;5.1.6  THE CPU TESTS THAT RELOCATE/REMAP WILL
28              ;IN THEIR ERROR TYPEOUTS:
29              ;ST.LA  START/ERROR (RES.)
30              ;XXXXXX YYYYYY  ZZZZZZ
31              ;
32              ;ST.LA  THE LOGICAL START OF THE RELOCATED TEST LOOP
33              ;XXXXXX (I.E. THE LAST LCALL SETUL)
34              ;
35              ;START  THIS NUMBER INDICATES WHERE THE RESIDENT COPY
36              ;YYYYYY OF THE TEST LOOP MAY BE FOUND IN THE LISTING
37              ;
38              ;ERROR  THIS NUMBER INDICATES WHERE IN THE RESIDENT
39              ;ZZZZZZ COPY OF THE LISTING THE ERROR CALL MAY BE FOUND
40              ;       (FOR SOME VALIDITY TRAP ERRORS THIS NUMBER
41              ;       MAY NOT APPEAR TO BE VALID.)
42              ;
43              ;5.1.7  THE CPU TESTS THAT RELOCATE WILL UPON
44              ;       DETECTING AN ERROR CHECK THE RELOCATED
45              ;       CODE TO VERIFY THAT IT IS INTACK. IF
46              ;       A DIFFERENCE IS FOUND THE FOLLOWING
47              ;       TYPEOUT IS INCLUDED IN THE ERROR
48              ;       TYPEOUT:
49              ;
50              ; RELOCATED CODE ERROR
51              ; EXPECTED ACTUAL ADDR-E  ADDR-A
52              ; XXXXX    YYYYYY  ZZZZZ   QQQQQ
53              ;
54              ;  WHERE,
55              ;       XXXXXX  IS THE UNRELOCATED CODE WORD
56              ;       YYYYYY  IS THE RELOCATED CODE WORD
57              ;       ZZZZZ   IS THE ADDR. OF THE UNRELOC. WORD
58              ;       QQQQQ   IS THE LOGICAL ADDR. OF RELOC. WORD
59              ;
```



```
01              ;5.2    ERROR ANALYSIS
02              ;       DUE TO THE INTERACTIVE NATURE OF
03              ;       THE TESTS INVOLVED, A SERIES OF
04              ;       ERROR TYPEOUTS WILL PROBABLY BE
05              ;       REQUIRED FOR ANALYSIS BEFORE A
06              ;       PROBLEM WILL BE ISOLATED.
07              ;       A RESTART AT 202 AND DELETION OF ALL
08              ;       BUT THE TEST THAT ORIGINALLY
09              ;       FAILED MAY HELP TO ISOLATE
10              ;       INTERACTIVE PROBLEMS AS FOLLOWS:
11              ;
12              ;5.2.1          IF THE TEST RUNS BY ITSELF THE PROBLEM
13              ;       IS INTERACTIVE-RE-ENABLE ONE OTHER TEST AT
14              ;       A TIME TO DETERMINE WHICH ONE IS THE PROBLEM.
15              ;       IF THE TEST DOES NOT RUN BY ITSELF
16              ;       RESORT TO SIMILAR BUT LOWER LEVEL TESTS
17              ;       FOR ISOLATION
18              ;5.3    PERTINENT MEMORY LOC'S TYPED
19              ;
20              ;5.3.1  CHECKERBOARD RAN
21              ;               THE AC'S AT ERROR WILL INDICATE:
22              ;               GOOD DATA- BAD DATA-LOGICAL ADDRESS
23              ;
24              ;               IN ADDITION THE FOLLOWING LOCATIONS ARE TYPED:
25              ;               CB.TK   TEST COUNTER
26              ;               0       GENERATE CHECKERBOARD
27              ;               1       DISTURB PASS
28              ;               2       CHECK PATTERN
29              ;               3       CHECKSUM THE # OF -1'S IN PATTERN
30              ;               CB.LC   STARTING LOGICAL ADDRESS OF "BEGIN"
31              ;                       RELOCATED TO SCRATCH
32              ;               CB.SE   AC3 AT ERROR CALL
33              ;
34              ;
35              ;5.3.2  SC MEMORY TEST
36              ;               THIS IS AN ISZ/DSZ TEST FOR SC-MEMORIES.
37              ;
38              ;               THE AC'S AT ERROR WILL INDICATE:
39              ;               ACTUAL-EXPECTED-LOGICAL ADDRESS
40              ;
41              ;               IN ADDITION THE FOLLOWING LOCATIONS ARE TYPED:
42              ;               MM.TK   ERROR NUMBER:
43              ;               0       PATTERN STORING ERROR(SHD BE -1)
44              ;               1       LOCATION NOT -1 BEFORE DOING ISZ
45              ;               2       ISZ DIDN'T SKIP
46              ;               3       LOCATION NOT EQUAL TO 0 AFTER ISZ
47              ;               4       DSZ SKIP ERROR
48              ;               5       DSZ TEST-LOCATION NOT -1 AFTER DSZ
49              ;               6       SAME AS 1, EXCEPT TESTING IN REV DIRECTION
50              ;               7       SAME AS 2, EXCEPT   "    "  "   "    "
51              ;               10      SAME AS 3, EXCEPT   "    "    "    "    "
52              ;               MM.SE   INSTRUCTION ADDRESS FOLLOWING ERROR CALL
53              ;               LOCATION  ADDRESS OF FAILING LOCATION(LOGICAL)
54              ;
55              ;
```

```
10011  N3MPL
01                ;5.3.3  ARITHMETIC TEST
02                ;       THE AC'S WILL BE TYPED AS THEY WERE AT THE
03                ;       TIME OF ERROR DETECTION
04                ;
05                ;       IN ADDITION THE FOLLOWING LOCATIONS ARE TYPED:
06                ;       AT.LC   STARTING ADDRESS OF ARITH IN SCRATCH
07                ;       AT.LO   LOW LIMIT OF SCRATCH AREA AFTER IT IS
08                ;               REMAPPED FOR EXECUTION
09                ;       AT.LA   AT.LC IN RELATION TO AT.LO
10                ;               (LOGICAL START OF ARITH AFTER REMAPPING)
11                ;       THE LAST THREE RANDOM NUMBERS GENERATED
12                ;       (SEE DISCUSSION OF ST.LA,ETC AT PARA.5.1.6)
13                ;
14                ;5.3.4  FLT PT TEST
15                ;
16                ;       AC0     GOOD DATA
17                ;       AC1     BAD DATA
18                ;       AC2     ADDRESS OF GOOD DATA DURING TEST EXECUTION
19                ;
20                ;       IN ADDITION THE FOLLOWING LOCATIONS ARE TYPED:
21                ;
22                ;       FP.LC   START OF LOCATIONS INCLUDING CURRENT
23                ;               TEST THAT IS IN SCRATCH(SEE FP.EN)
24                ;       FP.LO   SCRLO AFTER REMAPPING FOR EXECUTION
25                ;       FP.EN   END OF TEST OR START OF RANDOM DATA
26                ;               IN SCRATCH AREA
27                ;       PF803   AC3 AT TIME OF ERROR CALL (ADDR OF JSR )
28                ;
29                ;       (FOR THE REMAINDER OF THE OUTPUT SEE DISCUSION OF
30                ;       ST.LA ETC AT PARA. 5.1.6)
31                ;
32                ;
33                ;5.3.5  MUL/DIV TEST
34                ;
35                ;
36                ;               MULTIPLY DIVIDE FAILURES WILL INDICATE
37                ;               EITHER MUL FOR MULTIPLY OR DIV FOR DIVIDE
38                ;                IN ADDITION, THREE SETS OF AC'S ARE TYPED
39                ;                   ORIGINAL OPERANDS
40                ;                   HARDWARE RESULT (ASSUMED TO BE INCORRECT )
41                ;                   SOFTWARE RESULT (ASSUMED TO BE CORRECT )
42                ;
43                ;5.3.6  STACK ERROR TEST.
44                ;       THE STACK ERROR TEST OUTPUT IS AS FOLLOWS:
45                ;
46                ;       PROGRAM # NNN   STACK ERROR TEST.
47                ;       AC0     AC1     AC2
48                ;       NNNNNN  NNNNNN  NNNNNN
49                ;       SCRLO/HI NNNNNN NNNNNN  USER (A OR B)
50                ;
51                ;       (MEMORY ALLOCATION TABLE APPEARS HERE
52                ;       IF TEST IS RUNNING MAPPED.)
53                ;
54                ;       LOGICAL ADDRESS OF ERROR IS: NNNNN
55                ;
56                ;       THE USER MAY LOOK UP THE ADDRESS GIVEN AS THE LOGICAL
57                ;       ADDRESS OF THE ERROR IN THE LISTING.  THE TYPE OF ERROR
58                ;       AND THE MEANING OF THE ACS ARE GIVEN IN THE COMMENTS AT THAT
59                ;       POINT.
```

```
10012  N3MPL
01                ;5.3.7  NOVA  DISK
02                ;
03                ;       AC0     GOOD DATA
04                ;       AC1     BAD DATA
05                ;       AC2     ADRS. OF GOOD DATA (BAD IS AT AC2+4)
06                ;       ND.SA   MAY= ANY OF THE FOLLOWING
07                ;               A. THE ADRS. OF A RANDOM DATA CONTROL WORD
08                ;               (NDDW1 TO NDDW4) THE ERROR WAS IN ONE OF
09                ;               THE FIRST 4 WORDS IN THE BUFFER
10                ;               B. =AC2 ERROR WAS DISK STATUS
11                ;               C. A -# DATA ERR IS AT AC2+4
12                ;       NDDST   LOGICAL ADDR OF DATA START
13                ;       NDCST   LOGICAL CHANNEL ADDR OF DATA START
14                ;       NDSTA   LAST DISK STATUS
15                ;       NDADR   START SECTOR # OF THESE EXERCISED
16                ;       ND.CO   =1 OPERATION WAS A WRITE
17                ;       ND.CO   =0 OPERATION WAS A READ
```

```
!0013  N3MPL
01              ;5.3.8  6063/6064 DISK
02              ;
03              ; PD.8A IS THE KEY TO USING THE ERROR OUTPUT
04              ;       PD.8A   MAY=ANY OF THE FOLLOWING:
05              ;               =THE ADDRESS OF A RANDOM DATA CONTROL
06              ;               WORD(PDDW1 TO PDDW4).THE ERROR WAS IN
07              ;               ONE OF THE FIRST 4 WORDS IN THE BUFFER.
08              ;               = A NEGATIVE # ,DATA ERROR IS AT AC2+4
09              ;               = AC2 ,ERROR WAS A DISK STATUS ERR
10              ;                       IN WHICH CASE:
11              ;               AC0= DRV,TRK,SECTOR
12              ;                    FROM WORD 2 OF CMD QUEUE
13              ;               AC2= DISK DIC STATUS
14              ;
15              ; FOR DATA COMPARE ERRORS, THE AC'S
16              ; HAVE THE FOLLOWING INFORMATION:
17              ;       AC0     GOOD DATA
18              ;       AC1     BAD DATA
19              ;       AC2     ADDRESS OF GOOD DATA(BAD IS AT AC2+4)
20              ;
21              ; ALSO THE FOLLOWING INFORMATION IS OUTPUTTED:
22              ;       PDDST   DATA START IN CORE
23              ;       PD.CA   LOGICAL ADDR OF CHANNEL IN 1K'S OCTAL
24              ;       PDSTA   DISK STATUS(DIC)
25              ;       PDADR   DRIVE+TRACK+SECTOR(FIRST WORD OF QUEUE)
26              ;       PD.CO   0=READ,1=WRITE,2=DATA VERIFY
27              ;
28              ; ALSO IF THE ERROR OCCURS IN A READ OPERATION
29              ; THE FOLLOWING DATA IS PRINTED:
30              ;
31              ;   WRITE PDCST = XXXXX
32              ;       MD1     MD2     MD3     MD4
33              ;       GGGG    HHHH    JJJJ    KKKK
34              ;
35              ; WHERE,
36              ;       XXXXX = STARTING CHANNEL ADDRESS OF WRITE OPERATION
37              ;        GGGG = FIRST PHYS 1K USED IN DISK WRITE
38              ;        HHHH = 2ND     "    "  "   "   "    "
39              ;        JJJJ = 3RD     "    "  "   "   "    "
40              ;        KKKK = 4TH     "    "  "   "   "    "
41              ;
42              ; NOTE: UPON DETECTION OF AN ERROR THE TEST WILL
43              ;       RETRY THE OPERATION FOUR TIMES.
44              ;
45              ;*****************************************
46              ;       6063/64 DISK STATUS WORD
47              ;*****************************************
48              ; BIT(S)  MEANING        BIT(S)  MEANING
49              ; 0      ERROR FLG       10      DISK RDY
50              ; 3      BUS ENABLE      11      UNSAFE
51              ; 4-5    DISK CAPACITY   12      DATA LATE
52              ; 6      IDLE DONE       13      ECC
53              ; 7      WRITE PARITY    14      DATA VERIFY
54              ; 8      DCH ERROR       15      PAGE DONE
55              ; 9      READ/WRITE TIMEOUT
```

```
!0014  N3MPL
01              ;5.3.9  MOVING HEAD DISK TEST
02              ;
03              ; MH.8A IS THE KEY TO USING THE ERROR PRINT OUT.
04              ;       MH.8A   =ADDRESS OF A RANDOM DATA CONTROL WORD
05              ;               (MHDW1 TO MHDW4) THE ERR WAS IN ONE OF
06              ;               THE FIRST 4 WORDS IN THE BUFFER
07              ;               =-# ERR IS AT AC2+4
08              ;               =AC2  ERROR WAS DISK STATUS
09              ;                       IN WHICH CASE:
10              ;               AC0=DIB DISK
11              ;               AC1=DIC DISK
12              ;               AC2=DISK STATUS(DIA)
13              ;               AC3=DIC DISK
14              ; IN THE CASE OF A DATA COMPARE ERROR:
15              ;       AC0     GOOD DATA (SEE MH.8A)
16              ;       AC1     BAD DATA
17              ;       AC2     ADRS. OF GOOD DATA
18              ;               BAD IS AT AC2+4 IF MH.8A IS A -#
19              ; IN ADDITION THE FOLLOWING LOC'S ARE TYPED
20              ;       MHDST   DATA START IN CORE
21              ;       MHCST   DATA START FOR DCH MAP
22              ;       MHSTA   LAST DISK STATUS
23              ;       MHDOA   LAST DOA TO DISK
24              ;       MHDOC   LAST DOC TO DISK
25              ;
26              ; ALSO IF THE ERROR OCCURS IN A READ OPERATION
27              ; THE FOLLOWING DATA IS PRINTED:
28              ;   WRITE MHCST = XXXXX
29              ;       MD1     MD2     MD3     MD4
30              ;       GGGG    HHHH    JJJJ    KKKK
31              ;WHERE  XXXXX = STARTING CHANNEL ADDRESS OF WRITE OPERATION
32              ;        GGGG = FIRST PHYS 1K USED IN DISK WRITE
33              ;        HHHH = 2ND     "    "  "   "   "    "
34              ;        JJJJ = 3RD     "    "  "   "   "    "
35              ;        KKKK = 4TH     "    "  "   "   "    "
36              ; NOTE: UPON DETECTION OF AN ERROR THE TEST WILL
37              ;       RETRY THE OPERATION FOUR TIMES.
38              ;
39              ;*************************************************
40              ;       MOVING HEAD DISK STATUS WORDS
41              ;*************************************************
42              ; BITS  4047           4048,57   4231     6030,45 6067
43              ;  0    R/W DONE         SAME     SAME     SAME    DC DNE
44              ;  1    SEEK 0 DNE       SAME     SAME     SAME    CMD DNE 0
45              ;  2    SEEK 1 DNE       SAME     SAME     SAME    CMD DNE 1
46              ;  3    SEEK 2 DNE       SAME     SAME     SAME    CMD DNE 2
47              ;  4    SEEK 3 DNE       SAME     SAME     SAME    CMD DNE 3
48              ;  5    SEEK ON DRV 0    SAME   DUAL PRC DKT      DISKETTE
49              ;  6    SEEK ON DRV 1    SAME   SECT ERR V.S.     SEL.
50              ;  7    SEEK ON DRV 2    SAME   HEAD ERR N/A      BAD SECTOR
51              ;  8    SEEK ON DRV 3    SAME   ADDR ERR UNSAFE   UNSAFE
52              ;  9    DRIVE RDY        SAME     SAME     SAME    SAME
53              ; 10    SEEK ERR         SAME     SAME     SAME    SAME
54              ; 11    EOC ERR          SAME     SAME     SAME    SAME
55              ; 12    UNSAFE         ADDR ERR UNSAFE ADDR ERR   ADDR ERROR
56              ; 13    ECC ERR          SAME     SAME     SAME    SAME
57              ; 14    DATA LATE        SAME     SAME     SAME    SAME
58              ; 15    ERR              SAME     SAME     SAME    SAME
59              ;*************************************************
```

```
01              ;5.3.10 6060/61 DISK TEST
02              ;
03              ; Z8.8A IS THE KEY TO THE ERROR PRINT OUT:
04              ;       Z8.8A   =ADDRESS OF A RANDOM DATA CONTROL WORD
05              ;               (Z8DW1 TO Z8DW4) THE ERR WAS IN ONE OF
06              ;               THE FIRST 4 WORDS IN THE BUFFER
07              ;               =-# ERR IS AT AC2+4
08              ;               =AC2  ERROR WAS DISK STATUS
09              ;                     IN WHICH CASE:
10              ;               AC0=DIB DISK
11              ;               AC1=DIC DISK
12              ;               AC2=DIA DISK(STATUS)
13              ;               AC3=DIC DISK
14              ; IN THE CASE OF A DATA COMPARE ERROR:
15              ;       AC0     GOOD DATA (SEE Z8.8A)
16              ;       AC1     BAD DATA
17              ;       AC2     ADRS. OF GOOD DATA
18              ;               BAD IS AT AC2+4 IF Z8.8A IS A -#
19              ; IN ADDITION THE FOLLOWING LOC'S ARE TYPED
20              ;       Z8DST   DATA START IN CORE
21              ;       Z8CST   DATA START FOR DCH MAP
22              ;       Z8STA   LAST DISK STATUS
23              ;       Z8DOA   LAST DOA TO DISK
24              ;       Z8DOC   LAST DOC TO DISK
25              ;
26              ; ALSO IF THE ERROR OCCURS IN A READ OPERATION
27              ; THE FOLLOWING DATA IS PRINTED:
28              ;   WRITE Z8CST = XXXXX
29              ;       MD1     MD2     MD3     MD4
30              ;       GGGG    HHHH    JJJJ    KKKK
31              ;WHERE  XXXXX = STARTING CHANNEL ADDRESS OF WRITE OPERATION
32              ;        GGGG = FIRST PHYS 1K USED IN DISK WRITE
33              ;        HHHH = 2ND    "    "  "   "   "    "
34              ;        JJJJ = 3RD    "    "  "   "   "    "
35              ;        KKKK = 4TH    "    "  "   "   "    "
36              ;
37              ; NOTE: UPON DETECTION OF AN ERROR THE TEST WILL
38              ;       RETRY THE OPERATION FOUR TIMES.
39              ;
40              ;*********************************************
41              ;       6060/61 STATUS WORDS
42              ;*********************************************
43              ; BITS  DIA             DIB
44              ;  0    CNTL FULL       INVALID STATUS
45              ;  1    R/W DONE        DRV RESERVED
46              ;  2    SEEK 0 DONE     TRESPASSED
47              ;  3    SEEK 1 DONE     READY
48              ;  4    SEEK 2 DONE     BUSY
49              ;  5    SEEK 3 DONE     OFFSET
50              ;  6    PARITY ERROR    WRITE DISABLE
51              ;  7    ILLEGAL SECT    N/A
52              ;  8    ECC ERROR       ILLEGAL ADDR
53              ;  9    BAD SECTOR      ILLEGAL CMD
54              ; 10    CYL ERROR       PWR FAULT
55              ; 11    SURF/SECT ERR   PACK UNSAFE
56              ; 12    VERIFY ERROR    POSITIONER
57              ; 13    R/W TIMEOUT     CLK FAULT
58              ; 14    DATA LATE       WRITE FAULT
59              ; 15    RD/WRT FAULT    DRIVE DONE
60              ;*********************************************
```



```
01              ;5.3.11 MAGNETIC TAPE AND CASSETTE TEST
02              ;       AC0     GOOD DATA
03              ;       AC1     BAD DATA
04              ;       AC2     ADRS OF BAD DATA (GOOD IS AT AC2-4)
05              ;IN ADDITION THE FOLLOWING LOC'S ARE TYPED
06              ;       READ    INDICATES # BLOCKS IN BUFFER IF MODE=2
07              ;       W/DOB   LOGICALSTARTING ADRS. WHEN BLOCKS WERE WRITTEN
08              ;       LAST/DOB  LOGICALSTARTING ADRS. CURRENT OPERATION
09              ;       MODE    0=REWIND 1=WRITE 2=BACKSPACE OR READ
10              ;               DRIVE # WILL APPEAR IN BITS 4,5&6
11              ;       STATUS  LAST TAPE STATUS
12              ;       MT.EK   ERROR COUNTER STARTS AT 3 AND COUNTS
13              ;               DOWN FOR EACH REREAD
14              ;NOTE:  IF STATUS INDICATES TAPE ERR (BIT 0=1)
15              ;       THE CONTENTS OF AC0,1,AND 2 SHOULD BE IGNORED.
16              ;
17              ;5.3.12 LINE PRINTER
18              ;       NO ERROR TYPEOUTS.
19              ;       PRINTER OUTPUT MUST BE EXAMINED VISUALLY.
20              ;
21              ;5.3.14 DCU-50/200 TEST
22              ;
23              ;       THE AC'S WILL BE TYPED AS THEY WERE AT THE
24              ;       TIME OF ERROR DETECTION.
25              ;
26              ;       IN ADDITION THE FOLLOWING DATA IS TYPED:
27              ;       RANDOM DATA AC0,1,2
28              ;       DCLOR   LOGICAL START OF LOOP IN DCU
29              ;       DCLPK   LOOP COUNT
30              ;       DCLER   LOGICAL ERROR ADDRESS
31              ;       DC.LA   LOGICAL START OF TEST
32              ;       DC.LP   LISTING START OF LOOP
33              ;       ERROR   LISTING ADDR OF ERROR
```

```
01              ;5.4    SPECIAL CASE ERROR TYPEOUTS
02              ;
03              ;5.4.1  POWER FAIL INTERRUPT
04              ;       UPON DETECTION OF A POWER FAIL INTERRUPT
05              ;       THE LOGICAL ADRS. OF THE P.C. AT INTERRUPT
06              ;       WILL BE SAVED.
07              ;       IF AUTO-RESTART  IS ENABLED OR THE POWER
08              ;       FAIL WAS ONLY MOMENTARY, THE TEST WILL RE-
09              ;       START AS IN A START AT 204 AFTER TYPING
10              ;POWER FAIL @XXXXXX (WHERE XXXXXX IS THE PC AT INTR.)
11              ;
12              ;5.4.2  ILLEGAL SUPERVISOR CALL
13              ;
14              ;       UPON DETECTION OF A SUPERVISOR CALL
15              ;       WHICH DIDN'T MATCH THE LIST OF SUBROUTINES
16              ;       CALLS THE FOLLOWING MESSAGE WIIL BE TYPED:
17              ;
18              ;       ILLEGAL SUPER CALL AT XXXXXX
19              ;
20              ;       PROG#   NNN
21              ;
22              ;       AC'S    QQQQQQ  YYYYYY  ZZZZZZ
23              ;
24              ;       TTTTTT  WWWWWW  SSSSSS
25              ;
26              ;               WHERE XXXXXX IS THE LOGICAL ADDRESS OF THE
27              ;               SUPER CALL, TTTTTT IS AC3 CONTENTS
28              ;               AND WWWWWW IS THE PHYSICAL PAGE #,SSSSSS
29              ;               IS THE INSTRUCTION CAUSING THE SUPER-
30              ;               CALL.
31              ;
32              ;       NOTE:   IF THE ADDRESS TYPED IN THE ILLEGAL SUPERCALL
33              ;               WAS 000000 THEN THE PROGRAM WAS EXECUTING
34              ;               LOCATION 0.
```



```
01              ;5.4.3  I/O OR VALIDITY TRAP
02              ;       DEFER OR WRITE CHECK TRAP
03              ;         AN I/O,WRITE,DEFER OR VALIDITY TRAP
04              ;         OCCURED THAT WAS NOT FORCED BY ANY TEST
05              ;
06              ;         THE AC'S TYPED AFTER THE PROGRAM #
07              ;         ARE ASSOCIATED WITH THE FOLLOWING:
08              ;       AC0:    ADDRESS OF INSTR TRAPPED
09              ;       AC1:    VIOLATION DATA REGISTER CONTENTS
10              ;       AC2:    MAP STATUS BITS
11              ;
12              ;       STATUS BITS:
13              ;               0:      PROGRAM MAP ENABLE
14              ;               1:      DCH MAP ENABLE
15              ;               2:      PROGRAM MAP INHIBIT
16              ;               9:      SINGLE CYCLE WRITE PROTECT
17              ;               10:     SINGLE CYCLE MAP SELECT A/B
18              ;               11:     AUTOINDEX PROTECT
19              ;               12:     DEFER PROTECT
20              ;               13:     I/O PROTECT
21              ;               14:     WRITE PROTECT
22              ;               15:     PROGRAM MAP SELECT A/B
23              ;
24              ;5.4.4  INTERRUPT WAIT ELAPSED
25              ;       THE PERIPHERAL DEVICE ASSOCIATED WITH THE
26              ;       PROG. NUMBER TYPED HAS NOT RESPONDED WITH
27              ;       A PROGRAM INTERRUPT FOR AN EXTENDED
28              ;       PERIOD OF TIME. THE 2ND NUMBER TYPED
29              ;       SHOULD POINT AT THE INTERRUPT HANDLER
30              ;       FOR THE DEVICE THAT FAILED
```

```
                ;5.4.5  PARITY ERROR INTERRUPT
                ;
                ;       IN CASE OF A PARITY ERROR THIS TEST WILL
                ;       PRINT THE FOLLOWING INFORMATION:
                ;
                ;       PARITY ERROR
                ;       INTR    MADR    XMADR
                ;       XXXXX   YYYYY   ZZZZZ
                ;       WHERE,
                ;       XXXXX   ADDRESS(LOGICAL) WHERE INTERRUPT OCCURED
                ;       YYYYY   BIT 0 IS THE PARITY BIT
                ;               BITS 1-15 ARE THE MEM ADDR BITS 1-15
                ;       ZZZZZ   EXTENDED MEM ADDRESS BITS 0-2
                ;
                ;
                ;5.4.6  DATA CHANNEL PROTECTION ERROR
                ;
                ;       IF THE DATA CHANNEL PROTECTION FLAG IS
                ;       EVER FOUND TO BE SET(BUSY FLAG-DEV. #2)
                ;       THE FOLLOWING DATA WILL BE TYPED:
                ;
                ;       DCH  VIOLATION ERROR
                ;       PROG#   ERRORS
                ;       XX      YY
                ;
                ;       WHERE,
                ;       XX      IS THE PROGRAM NUMBER EXECUTING WHEN
                ;               THE ERROR WAS DETECTED.
                ;       YY      IS THE NUMBER OF DCH ERRS DETECTED SINCE
                ;               LAST REPORTED.
```



```
                ;6.0    DIAGNOSTIC LINKER
                ;
                ;6.1    PROGRAM INITIALIZE
                ;THE DIAGNOSTIC LINKER INITIALIZES ITSELF
                ;AND INDIVIDUAL TESTS IN THE FOLLOWING
                ;SEQUENCE:
                ;   1. SYSTEM IS RESET, MAP OPTION IS
                ;      DETERMINED TO EXIST OR NOT EXIST
                ;      AND SWITCHES ARE SET UP
                ;      ACCORDINGLY
                ;   2. ANY OTHER NECESSARY CONSTANTS
                ;        ARE INITIALIZED
                ;      (MEM ALLOCATION TABLES)
                ;   3. INTERRUPT VECTOR TABLES ARE SET UP TO
                ;      PROCESS UNEXPECTED DEVICE INTERRUPTS
                ;   4. MEMORY IS SIZED
                ;      FROM 0 TO 128K AND BUILDS AN 8 WORD
                ;      BIT MAP OF EXISTING CONTIGUOUS
                ;      MEMORY
                ;
                ;   5. THE EXIST MAP IS MOVED TO THE
                ;      AVAILABLE MAP AND EACH BIT
                ;      CORRESPONDING TO 1K OF UITILIZED
                ;      MEMORY IS REMOVED FROM THE MAP
                ;      SO THAT IT WILL NOT BE ASSIGNED
                ;      AS A SCRATCH AREA TO ANY TEST.
                ;      (INCLUDES PROGRAM STORAGE, MEMORY ALLOC.
                ;      TABLES, INTERRUPT MASKS AND STACK AREA AND
                ;      THE LAST 1K OF MEMORY TO PRESERVE THE
                ;      BINARY LOADER)
                ;   6. EACH TEST IS ENTERED IN SEQUENCE AT ITS
                ;      INIT. ENTRY POINT. OPTION TESTS DETERMINE
                ;      IF THE DEVICE THEY ARE ASSOC. WITH EXISTS
                ;      OR NOT AND PASS INTERRUPT SERVICE PARAM'S
                ;      TO THE LINKER.
                ;        (DEV#, MASK AND INTERRUPT SERVICE
                ;      ADDRESS)
                ;   7. LINKER THEN TYPES THE SYSTEM SIZE
                ;      INFORMATION ALONG WITH THE PROGRAM
                ;      RUN LIST AND WILL ALLOW THE OPERATOR
                ;      TO SELECT OR DELETE SPECIFIC TESTS
                ;      IF START WAS 202 .
                ;   8. AFTER STARTING, THOSE TESTS THAT HAVE
                ;   "SIZED" THEIR SUBSYSTEM FOR SPECIFIC
                ;   PARAMETERS TYPE AN INDICATION OF THE PARAMETERS
                ;   THEY DETERMINED TO EXIST.(SEE THE NOVA DISK,6060/61 DISK,
                ;   6063/64 DISK, AND MOVING HEAD DISK TEST DESCRIPTIONS.)
```

```
10021  N3MPL
01                ;6.2    PROGRAM RUN
02                ;       ONCE THE LINKER HAS COMPLETED ALL
03                ;       INITIALIZATION THE FOLLOWING SERIES
04                ;       OF OPERATIONS IS LOOPED THROUGH
05                ;
06                ;          1. LINKER RANDOMLY SELECTS ONE OF
07                ;             THE INDIVIDUAL TESTS UNTIL IT
08                ;             FINDS ONE THAT IS NOT WAITING
09                ;             FOR INTERRUPT (WAIT IS BIT 0 OF
10                ;             THE THIRD WORD IN TEST=1) AND THAT
11                ;             THE NEXT RANDOM NUMBER FALLS WITHIN
12                ;             ITS ENTER LIMITS
13                ;
14                ;          2. IF THE MAP OPTION EXISTS, ALL LOGICAL PAGES
15                ;             EXCEPT PAGE 0 ARE ACCESS PROTECTED WITH
16                ;             THE PHYSICAL AREA OF THE SELECTED TEST
17                ;             MAPPED TO ITSELF AND ANY ASSIGNED
18                ;             SCRATCH AREA MAPPED TO START AT 1K
19                ;             ABOVE THE TEST, MEMORY LOCATIONS SCRLO
20                ;             AND SCRHI (SCRATCH LOW AND HIGH) ARE
21                ;             SET TO INDICATE THE LIMITS OF
22                ;             THE SCRATCH AREA AVAILABLE TO THE TEST.
23                ;
24                ;          3. DATA CHANNEL LIMITS (DCHLO AND DCHHI)
25                ;             ARE CALCULATED AND ENTERED
26                ;
27                ;          4. THE SELECTED TEST IS ENTERED AT
28                ;             ITS SPECIFIED EXECUTE ENTRY POINT
```

```
10022  N3MPL
01                ;6.4    INDIVIDUAL TEST DESCRIPTIONS
02                ;
03                ;6.4.1  CHECKERBOARD RAN
04                ;
05                ;THIS MEMORY CHECKER BOARD TEST IS A SUBSET OF OTHER MEMORY
06                ;CHECKERBOARDS.  A COMPLETE TEST OF AN AVAILABLE SCRATCH
07                ;AREA IS COMPRISED OF THE FOLLOWING SEQUENCE:
08                ;
09                ;CB.TK=0        ;REQUEST 1 TO 20K OF SCRATCH, RANDOMLY RE-
10                ;               ;LOCATE THE EXECUTE PORTION OF CHECKERBOARD
11                ;               ;INTO SCRATCH AND GENERATE THE CHECKERBOARD
12                ;               ;PATTERN
13                ;
14                ;CB.TK=1        ;DISTURB PASS-COMPLIMENT A SINGLE BIT IN EACH
15                ;               ;OF THE FIRST 16 WORDS OF SCRATCH, SHUFFLE THESE
16                ;               ;WORDS 16 TIMES SUCH THAT THEY END UP IN THEIR
17                ;               ;ORIGINAL POSITION, RE-COMPLIMENT THE SINGLE
18                ;               ;BIT IN EACH WORD-PROCEED WITH EACH GROUP OF
19                ;               ;16 WORDS UNTIL ALL MEMORY HAS BEEN EXERCISED.
20                ;
21                ;CB.TK=2        ;CHECK PASS-COMPARE EACH WORD IN SCRATCH WITH
22                ;               ;THE PATTERN EXPECTED
23                ;
24                ;CB.TK=3        ;FAST CHECKSUM MEMORY TO ENSURE THAT ALL DATA
25                ;               ;IS INTACT (RETURNS TO CHECK PASS IF CHECK-
26                ;               ;SUM DOES NOT AGREE.)
27                ;
28                ;
29                ;6.4.2  DCU - 50 TEST
30                ;
31                ;       THE MULTI-PROGRAMMING RELIABILITY DCU TEST RUNS
32                ;       AN ARITHMETIC TEST VIA THE DCH .
33                ;       THE DCU INTERRUPTS THE CPU
34                ;       WHEN EITHER IT HAS COMPLETED THE TEST OR
35                ;        FINDS AN ERROR.
36                ;       THIS TEST WILL AUTOSIZE FOR DCU-50/200 AT DEVICE 64.
37                ;       IF DCU-50 IS SETUP FOR OTHER DEVICE CODE, PATCH
38                ;       THE DEVICE CODE INTO LOCATION DCUDV. IF DCU-50/200
39                ;       DOESN'T RESPOND TO DEVICE CODE THEN TEST IS DELETED
40                ;       AUTOMATICALLY.
41       015306           LOC.=DCUDV
```

```
01              ;6.4.3  SC MEMORY TEST
02              ;
03              ;THIS MEMORY TEST DOES A READ/MODIFY/WRITE TO THE AVAILABLE
04              ;SCRATCH AREA USING AN "ISZ" INSTRUCTION.TEST IS BROKEN INTO THE
05              ;FOLLOWING CHECKS:
06              ;
07              ;       MM.TK= 0        WRITE INTO EACH MEMORY LOCATION A MINUS
08              ;                       ONE STARTING AT SCRLO AND ENDING AT SCRHI
09              ;                       VERIFYING EACH GOT THERE.
10              ;
11              ;       MM.TK= 1        READ A LOCATION BEFORE DOING THE ISZ
12              ;                       TO VERIFY IT HASN'T BEEN DISTURBED.
13              ;
14              ;       MM.TK= 2        ISZ DION'T SKIP
15              ;
16              ;       MM.TK= 3        LOCATION NOT 0 AFTER ISZ
17              ;
18              ;       MM.TK= 4        DSZ SKIPPED-ERROR
19              ;
20              ;       MM.TK= 5        DSZ TST- LOCATION NOT -1 AFTER DSZ
21              ;
22              ;       MM.TK= 6        SAME AS 1, EXCEPT TESTING IN THE REVERSE
23              ;                       DIRECTION
24              ;
25              ;       MM.TK= 7        SAME AS 2, EXCEPT TESTING IN THE REVERSE
26              ;                       DIRECTION.
27              ;
28              ;       MM.TK= 10       SAME AS 3, EXCEPT TESTING IN THE REVERSE
29              ;                       DIRECTION.
```



```
01              ;6.4.4  ARITHMETIC TEST
02              ;
03              ;THE MULTIPROGRAMMING RELIABILITY ARITHMETIC TEST WAS
04              ;DERIVED FROM THE STAND ALONE ARITHMETIC TEST.  THIS TEST
05              ;REQUIRES 2K OF SCRATCH FOR EXECUTION.  THE EXECUTE POR-
06              ;TION OF THE TEST IS RANDOMLY RELOCATED WITHIN AVAILABLE
07              ;SCRATCH.  IF THE SYSTEM IS MAPPED, (HAS AN MMU) THE
08              ;SCRATCH AREA IS RANDOMLY REMAPPED TO SOME OTHER LOGICAL AD-
09              ;DRESS FOR EXECUTION.  AT THE END OF EACH EXECUTION PASS SCRATCH
10              ;AREA IS RANDOMLY RELEASED OR HELD.  IF HELD, THE NEXT TIME
11              ;THE TEST IS ENTERED, THE EXECUTABLE PORTION OF THE TEST WILL
12              ;AGAIN BE RANDOMLY RELOCATED WITHIN SCRATCH FOR EXECUTION.
13              ;
14              ;6.4.5  FLOATING POINT TEST
15              ;
16              ;    THE NOVA MULTIPROGRAMMING RELIABILITY TEST
17              ;PERFORMS A LOAD AND STORE SINGLE PRECISION WITH
18              ;COMPARE TEST AND A LOAD AND STORE DOUBLE PRECISION
19              ;WITH COMPARE TEST.
20              ;    THIS TEST IS ENTERED IF A DEVICE CODE OF 76 WAS
21              ;FOUND DURING THE INITIALIZATION PORTION.
22              ;  A RANDOM SCRATCH AREA OF 1 - 32 K IS ASSIGNED FOR
23              ;FOR EXECUTION. THE EXECUTE PORTION OF THE TEST IS
24              ;RANDOMLY RELOCATED WITHIN THE AVAILABLE SCRATCH AREA.
25              ;  IF THE SYSTEM IS MAPPED, (HAS A MMPU) THE SCRATCH
26              ;AREA IS RANDOMLY REMAPPED TO SOME OTHER LOGICAL
27              ;ADDRESS FOR EXECUTION. AT THE END OF EACH EXECUTION
28              ;PASS SCRATCH AREA IS RANDOMLY RELEASED FOR OR HELD.
29              ;IF HELD, THE NEXT TIME THE TEST IS ENTERED, THE
30              ;EXECUTABLE PORTION OF THE TEST WILL AGAIN BE RANDOMLY
31              ;RELOCATED WITHIN SCRATCH FOR EXECUTION.
32              ;
33              ;6.4.6  MUL/DIV TEST
34              ;  THE NOVA MULTIPROGRAMMING MULTIPLY/DIVIDE TEST
35              ;PERFORMS A TRIAL INSTRUCTION TO DETERMINE
36              ;IF THE MULTIPLY/DIVIDE OPTION IS INSTALLED.
37              ;  THIS TEST WAS DERIVED FROM THE STAND ALONE
38              ;  MUL/DIV TEST.
39              ;  NO MEMORY REALLOCATING IS DONE IN THIS TEST.
40              ;
41              ;6.4.7 STACK ERROR TEST
42              ;  THE STACK ERROR TEST IS DESIGNED TO EXERCISE THE
43              ;STACK OVERFLOW FUNCTION OF THE NOVA.  IT SHOULD
44              ;BE NOTED THAT THIS TEST DOES NOT SET UP THE STACK.
45              ;THE TEST EXPECTS THE LINKER TO HAVE ESTABLISHED THE
46              ;STACK BY THE TIME THE TEST IS CALLED.
47              ;
48              ;WHEN EXECUTED THE STACK TEST FORCES A STACK OVERFLOW
49              ;AND THEN CHECKS THE FOLLOWING PARAMATERS:
50              ;    VERIFY THAT THE FAULT OCCURED.
51              ;    VERIFY THE DATA PUSHED ON THE STACK TO TRIGGER FAULT.
52              ;    VERIFY CORRECT RETURN ADDRESS PLACED IN LOC. 0.
53              ;    VERIFY STACK POINTER CORRECT AFTER PUSH.
54              ;IF ANY OF THESE PARAMATERS DOES NOT CHECK OUT AN ERROR
55              ;IS ANNOUNCED.
```

```
01              ;6.4.8  6063/64  DISK
02              ;
03              ;DURING INITIALIZATION THE TEST CHECKS FOR THE
04              ;EXISTANCE OF A DISK CONTROLLER AND THEN CHECKS
05              ;FOR THE EXISTANCE OF ANY/ALL DRIVES. A WRITE
06              ;BUFFERS COMMAND IS USED TO SIZE FOR AVAILABLE
07              ;DRIVES.
08              ;
09              ;THE TESTING OF EACH AVAILABLE DRIVE IS CONTROLLED
10              ;BY SELECTION OF ONE OF THREE OPERATION TABLES
11              ;PER DRIVE. EACH CONTROL TABLE IS 13 WORDS IN
12              ;LENGTH. THE FIRST WORD CONTAINS THE TRACK NUMBER
13              ;(RANDOMLY SELECTED), THE SECOND WORD CONTAINS THE
14              ;STARTING SECTOR AND NUMBER OF SECTORS USED. THE
15              ;STARTING SECTOR IS RANDOMLY SELECTED AND THE NUMBER
16              ;OF SECTORS IS DETERMINED BY THE AMOUNT OF SCRATCH
17              ; AREA ASSIGNED TO THE DISK TEST WHEN THE DISK
18              ;IS WRITTEN.
19              ;THE THIRD WORD IS A RETRY COUNTER
20              ;THE FOURTH WORD CONTAINS THE FIRST WORD OF THE
21              ;COMMAND QUEUE,I.E.,DRIVE+TRACK+SECTOR.
22              ;THE FIFTH THRU EIGHTH WORDS ARE THE RANDOM DATA
23              ;WORDS USED TO CREATE THE TEST PATTERN.(THEY REPEAT
24              ;EVERY FOUR WORDS)
25              ;THE NINTH WORD IS THE CHANNEL ADDR. USED IN
26              ;WRITING TO THE DISK. WORDS TEN THRU THIRTEEN
27              ;ARE THE PHYSICAL MEMORY 1K'S USED TO WRITE
28              ;TO THE DISK.
29              ;
30              ;UPON ENTERING FOR INITIAL EXECUTION, THE TEST ATTEMPS TO
31              ;ACQUIRE 1-4K OF SCRATCH AREA. THE TEST THEN RANDOMLY
32              ;SELECTS A DATA STARTING ADDRESS AFTER THE FIRST
33              ;256 WORDS IN SCRATCH. THE FIRST 256 WORDS ARE RESERVED
34              ;FOR THE COMMAND QUEUE.
35              ;THE TEST THEN SELECTS ONE OF THE AVAILABLE
36              ;DRIVES AND ONE OF THE THREE OP TABLES FOR THAT DRIVE.
37              ;IF THE FIRST WORD OF THE TABLE IS NON-ZERO(INDICATING
38              ;THE THE TRACK # IN WORD #1,STARTING AT THE SECTOR IN
39              ;BITS 11-15 OF THE SECOND WORD,FOR THE # OF SECTORS
40              ;SPECIFIED BY THE BITS 3-7 IN THE SECOND WORD, RANDOM DATA
41              ;HAS BEEN WRITTEN THAT IS EQUAL TO THE CONTENTS OF WORDS
42              ;4 THRU 7 F THE OP TABLE) THEN THE NEXT OPERATION OF
43              ;READ OR DATA VERIFY IS RANDOMLY SELECTED.
44              ;IF THE FIRST WORD OF THE TABLE IS ZERO, A TRACK IS
45              ;SELECTED WHICH IS NOT CURRENTLY IN AN OP TABLE,
46              ;AND A STARTING SECTOR # IS RANDOMLY SELECTED SUCH THAT
47              ;THE # OF SECTORS WRITTEN WILL NOT MAKE THE SECTOR #
48              ;FIELD OVERFLOW INTO THE TRACK FIELD.(I.E.,THE STARTING
49              ;SECTOR FALLS BETWEEN 0 AND 32-# OF SECTORS TO BE WRITTEN)
50              ;THE DATA PATTERN IS GENERATED IN SCRATCH AREA AND
51              ;A WRITE OPERATION IS SELECTED.
52              ;
```



```
01              ;6.4.7  6063/64  DISK (CONTINUED)
02              ;AFTER SELECTION OF THE OPERATION TO BE PERFORMED,
03              ;A COMMAND QUEUE IS GENERATED IN THE FIRST 256 WORDS
04              ;OF SCRATCH(FIVE WORDS PER SECTOR). THE FIRST WORD
05              ;CONTAINS THE DRIVE,TRACK AND SECTOR TO BE EXERCISED.
06              ;THE SECOND WORD CONTAINS THE COMMAND AND EXTENDED
07              ;ADDRESS BITS OF THE DATA ADDRESS. WORD # 3 CONTAINS
08              ;THE THE LOWER 16 BITS OF THE LOGICAL ADDRESS OF THE
09              ;DATA . WORD # 4 IS USED BY THE DISK TO STORE THE
10              ;STATUS WORD. WORD # 5 IS NOT CURRENTLY USED.
11              ;
12              ;THE LAST QUEUE BLOCK CONTAINS A HALT BIT IN WORD
13              ;# 2 WHICH TERMINATES THE OPERATION AFTER
14              ;COMPLETION OF THE SECTOR AND CAUSES THE DISK
15              ;TO INTERRUPT THE CPU.
16              ;THE DISK STATUS IS CHECKED AT THE TIME OF THE INTERRUPT
17              ;AND IF OK THE SCRATCH AREA IS VERIFIED. A STATUS ERROR OF
18              ;DATA LATE WILL CAUSE THE TEST TO RETRY THE OPERATION
19              ;UP TO FOUR TIMES. AS THE
20              ;DATA COMPARES ARE PERFORMED A NEGITIVE COUNT IS
21              ;STORED IN THE CHECKED LOCATION TO CLEAR THE BUFFER.
22              ;
23              ;6.4.9  NOVA DISK TEST
24              ;
25              ;WHEN ENTERED FOR ITS INITIALIZATION PASS THE
26              ;NOVA DISK TEST SIZES THE DISK SYSTEM.  IT DOES THIS BY
27              ;INITIATING A READ AT THE HIGHEST ADDRESS OF
28              ;EACH AVAILABLE DISK UNTIL IT NO LONGER RECEIVES A
29              ;NONEXISTENT DISK BIT IN THE STATUS.
30              ;THIS "HIGH DISK ADRS." IS TYPED THE FIRST
31              ;TIME THE NOVA DISK TEST IS ENTERED.
32              ;DURING RUNNING THE DISK TEST KEEPS 3 RANDOM
33              ;CONTROL TABLES THAT CONTAIN THE FOLLOWING INFO:
34              ;       A. THE START SECTOR # OF 16 RANDOMLY SELECTED
35              ;          SECTORS.  (THIS # IS RIGHT JUSTIFIED 4 BITS)
36              ;       B. A START SECTOR RANDOMLY SELECTED WITHIN 16
37              ;       C. THE NUMBER OF SECTORS TO BE WRITTEN/READ
38              ;       D. 4 RANDOM DATA WORDS THAT KEY THE PATTERN
39              ;          (THESE 4 WORDS REPEAT EVERY 4 WORDS)
40              ;THE TEST OPERATES OFF THESE RANDOM CONTROL TABLES
41              ;IN A MANNER SIMILAR TO THAT DESCRIBED FOR THE INDIVIDUAL
42              ;DISKS IN THE MOVING HEAD DISK TEST.
```

```
10029  N3MPL
01              ;6.4.12 LINE PRINTER TEST
02              ;
03              ;THE NON-DCH LINE PRINTER TEST RANDOMLY PRINTS 10 TO 60 LINES
04              ; PER PAGE WITH RANDOM STALLS EVERY 1 TO 9 LINES.
05              ;EACH LINE OF PRINT CONSISTS OF THE CHARACTERS SPACE
06              ;(40) TO Z (132).  THE TEST FILLS THE PRINT BUFFER UNTIL THE
07              ;FIRST PRINT CYCLE STARTS.  CONTINUATION OF PRINTING UNTIL
08              ;RANDOM STALL IS THEN RUN OFF INTERRUPTS FROM THE PRINTER.
09              ;  THE DCH-LINE PRINTER TEST ASSIGNS 1 TO 2K OF
10              ;SCRATCH AND ASSIGNS IT TO THE DCH A MAP. IT THEN RANDOMLY
11              ;CHOSES A STARTING ADDRESS 0 TO 63 WORDS INTO THE SCRATCH AREA
12              ; NEXT THE TEST CHOSES 10 TO A MAXIMUM OF 60 LINES TO PRINT.
13              ;THE PATTERN PRINTED CONSISTS OF THE CHARACTERS SPACE(40) TO
14              ; ](135). A TAB RUNAWAY ERROR WILL RESULT IN A PROGRAMED HALT.
15              ;
16              ;6.4.13 REAL TIME CLOCK
17              ;
18              ;THE REAL TIME CLOCK IS RUN AT 1K HERTZ.  RUNTIME ALONG
19              ;WITH ACCUMULATED ERROR COUNT ARE PRINTED AT 5 MINUTES
20              ;15 MINUTES, 30 MINUTES AND EVERY 30  MINUTES OF RUNTIME
21              ;THEREAFTER.  THIS TYPEOUT ALSO OCCURS AFTER EVERY ERROR
22              ;TYPEOUT OR IF TTY KEY 4 IS TYPED.
23              ;
24              ;6.4.14 TELETYPE TEST
25              ;
26              ;THE TELETYPE TEST PRINTS A SINGLE LINE CONSISTING OF THE
27              ;CHARACTERS SPACE TO Z.  THE TEST WILL ALSO ECHO CHARACTERS
28              ;AS TYPED.
```

```
10030  N3MPL
01              ;7.0 ODT EDITOR
02              ;7.1    REQUESTING THE ODT EDITOR
03              ;       TO ENTER THE ODT TYPE A CONTROL O ON
04              ;       THE TTI. THIS CAN BE DONE AT ANY POINT IN THE
05              ;       PROGRAM.
06              ;7.2    RESPONSE
07              ;       ON ENTERING THE ODT A CARRIGE RETURN, LINE FEED
08              ;       AND AN @ IS TYPED ON THE TTO.
09              ;
10              ;7.3 CONVENTIONS AND SYMBOLS IN COMMAND LINES
11              ; ------------------------------------
12              ;
13              ; CR    PRESSING THE RETURN KEY IS REPRESENTED BY CR .
14              ;
15              ; LF    PRESSING THE LINE FEED KEY IS REPRESENTED BY LF .
16              ;
17              ;?      PRESSING AN ILLEGAL KEY CAUSES THE ODT TO RESPONSE WITH
18              ;       A ?.
19              ;
20              ;@      ODT IS READY AND AT YOUR SERVICE.
21              ;
22              ;
23              ;7.4 COMMAND STRUCTURE
24              ;  -----------------
25              ;
26              ;       AN ODT COMMAND HAS THE GENERAL FORMAT:
27              ;
28              ;               [ARGUMENT][COMMAND]
29              ;
30              ;       ARGUMENT MAY BE ONE OF THE FOLLOWING:
31              ;
32              ;       ADR     AN OCTAL ADDRESS OR AN EXPRESSION OF THE FORM:
33              ;               X+X+X....
34              ;               WHERE EACH X IS AN OCTAL INTEGER, SEPARATED
35              ;                FROM THE FOLLOWING X BY EITHER +(PLUS)
36              ;               OR -(MINUS). LEADING ZEROS NEED NOT BE TYPED.
37              ;
38              ;       N       AN OCTAL INTEGER.
39              ;
40              ;       A COMMAND IS A SINGLE TELETYPE CHARACTER
41              ;
42              ;       CHARACTERS USED TO OPEN/CLOSE LOCATIONS INCLUDE:
43              ;       "/"  "CR"  "LF"  "^"
44              ;
45              ;       CHARACTERS USED TO ENTER/EXIT ODT INCLUDE:
46              ;       "^O"(CTRL O)  "R"  "P"
47              ;
48              ;       CHARACTERS USED TO MODIFY CURRENT ARGUMENTS ARE:
49              ;       "RUBOUT"  "+" "-" AND THE INTEGERS 0 TO 7
50              ;
51              ;       THE CHARACTER "=" ALLOWS THE CURRENT ARGUMENT TO BE
52              ;       EXAMINED WITHOUT OPENING OR CLOSING THE CURRENT LOC.
53              ;       CHARACTERS USED TO MANIPULATE THE NOVA 3 MAP INCLUDE:
54              ;       "M"  "A"  "B"  "U"  "E" "T"
55              ;
```

```
 0001  NMDU2    AOS ASSEMBLER REV 02.02         14:46:46 12/28/78
01
02
03
04
05
06
07              ;*********************************************************************
08              ;
09              ; NAME: N4MDU2.TX                          PART NUMBER: 097-001135
10              ;
11              ;
12              ; DESCRIPTION: NOVA 4 MEMORY DIAGNOSTIC / UNMAPPED / PART 2
13              ;
14              ;
15              ; REVISION HISTORY:
16              ;
17              ;       REV.            DATE
18              ;
19              ;       00              12/27/78
20              ;
21              ;
22              ; COPYRIGHT (C) DATA GENERAL CORPORATION, 1978
23              ; ALL RIGHTS RESERVED.
24              ;*********************************************************************
```

N4 MDU

```
10002  NMDU2
01       000000          .00     0
02
03                                      CONTENTS
04                                      ========
05
06
07              I. GOAL
08
09              II. DESIRED ATTRIBUTES
10
11              III. WHAT THE PROGRAM WILL DO AND WHY
12
13              IV. DESIGN HIGHLIGHTS
14
15              V. KNOWN ANOMALIES
16
17
18              TABLE 1- TEST NAMES' BIT POSITIONS
19
20              TABLE 2- CONTROL FLAGS' DEFAULT SETTINGS
21
22              TABLE 3- PROGRAM MONITOR COMMANDS
23
24
25              FIGURE 1- FAULT TYPES VS. MEMORY TEST PATTERNS
26
27              FIGURE 2- ERROR LOG OUTPUT
28
29              FIGURE 3- MEMORY TOPOLOGY, 4K RAMS
30
31
32                                      LAST MODIFICATION:      12/27/78    15:06
```

I. GOAL

PROVIDE A MEANS OF TESTING THE FIRST 8K WORDS OF MEMORY (MOS DYNAMIC 4K RAMS) ON AN UNMAPPED NOVA 4.

II. DESIRED ATTRIBUTES

A) SPEED- ENTIRE PROGRAM SHOULD RUN NO LONGER THAN 10-15 MINUTES IN AUTO MODE.

B) THOROUGHNESS- ALL MEMORY FAULT TYPES SHOULD BE TESTED.

C) STRENGTH- EACH MEMORY FAULT TYPE SHOULD BE CHECK BY A MEMORY TEST PATTERN OF AT LEAST MEDIUM STRENGTH. THE STRONGER A TEST, THE MORE LIKELY IT IS TO FIND A BUG. IN A SIMILAR VEIN, "WEAK" TESTS FIND HARD ERRORS, "STRONG" TESTS FIND SOFT (I.E., INTERMITTENT, INTERACTIVE) ERRORS.

D) DTOS COMPATIBLE.

E) BE USED UNDER ANY OF THE FOLLOWING POSSIBLE CONFIGURATIONS:

CONFIG=0	(1) 16K BOARD
CONFIG=1	(1) 32K BOARD
CONFIG=2	(2) 16K BOARDS

F) OPTIONS- UNDER MANUAL CONTROL, THE USER MAY OPT FOR TESTS WHICH COVER LESS IMPORTANT FAULT TYPES AND/OR TAKE A LONG TIME TO RUN. HE CAN ALSO ENABLE SPECIAL FEATURES WHICH AID IN THE LOCATION OF DIFFICULT ERRORS.

G) IF AN ERROR IS FOUND, PRINT OUT THE EXACT LOCATION OF THE BAD CHIP.

H) PROGRAM LENGTH WILL BE 8K, AND RESIDE IN LOCATIONS 20000 TO 37777.

I) AN ERROR LOG BE KEPT, WHICH CONTAINS A HISTORY OF ALL ERRORS ENCOUNTERED.

III. WHAT THE PROGRAM WILL DO AND WHY

A) IN AUTO MODE

DEFINITION: IN AUTO MODE, ALL CONTROL FLAGS (CF'S) ARE SET TO DEFAULT. SEE TABLE 2 FOR DEFAULT SETTINGS. 'AUTO MODE' IS RUN BY ISSUING ANY AUTO MODE COMMAND IN DTOS (E.G., LOAD.NN MEMD, AUTO). ALSO, AUTO MODE CAN BE SIMULATED BY RUNNING UNDER DTOS MANUAL MODE BUT USING THE DEFAULT SETTINGS FOR ALL CF'S.

1. SIZE MEMORY- RUN A TOP DOWN MEMORY SIZING ROUTINE TO DETERMINE THE CONFIGURATION OF THE MACHINE. THE THREE POSSIBLE BOARD CONFIGURATIONS ARE LISTED IN II. E). THIS IS NECESSARY FOR CONTROL OF VARIOUS PARAMETERS WITHIN THE PROGRAM, ESPECIALLY IN DETERMINING THE EXACT LOCATION OF ANY ERRORS (MODULE, BANK, AND BIT).

2. RUN THE SPECIFIED TESTS

FIVE TESTS HAVE BEEN CHOSEN THAT MOST OPTIMALLY MEET THE DESIRED ATTRIBUTES: 1) MODIFIED DATA=ADDRESS; 2) STUCK ADDRESS BIT; 3) MARCHING 1/0; 4) GALLOPING COLUMN; AND 5) GALLOPING ROWS.

THE TESTS WILL BE RUN IN THAT ORDER. THIS WAY, IF THE FAULT IS A HARD ERROR, THEN ONE OF THE FIRST THREE QUICK TESTS WILL FIND IT. IF NOT, THE GALLOPING TESTS WILL THEN BE RUN. SEE TABLE 1 FOR APPROXIMATE TEST TIMES FOR EACH PATTERN.

THE MODIFIED DATA=ADDRESS TEST WILL BE RUN IN TWO PHASES. IN THE FIRST PHASE OF THE MODIFIED DATA=ADDRESS TEST, THE DATA WORD WRITTEN TO AND SUBSEQUENTLY READ FROM THE MEMORY LOCATION WILL BE THE LOWER 14 BITS OF ITS 20-BIT ADDRESS. ON THE SECOND PHASE, THE COMPLEMENT OF THE LOWER 14 BITS OF THE 20-BIT ADDRESS WILL BE WRITTEN TO/READ FROM THE LOCATIONS.

THE GALLOPING TEST PATTERNS WILL BE RUN IN 8K CHUNKS. EXTENSIVE USE OF THE MMPU1 FEATURE AND MICROCODE SUPPORT IS MADE.

THESE FIVE TESTS COVER THE FOLLOWING SIX FAULT TYPES: 1) FAULTY ADDRESS DECODERS; 2) INTERACTIVE COUPLING BETWEEN CELLS; 3) REFRESH SENSITIVITY; 4) SLOW ACCESS TIME; 5) SENSE AMPLIFIER RECOVERY; 6) CELL FUNCTIONALITY (CAN THE CELL HOLD WHAT IT'S TOLD TO); AND 7) STUCK ADDRESS BITS AT THE CHIP LEVEL.

FIGURE 1 DETAILS WHICH TESTS COVER WHAT FAULT TYPES.

EACH TEST WILL BE RUN SEQUENTIALLY DEPENDENT ON ERROR CONDITIONS. SPECIFICALLY, TEST 1 IS RUN. IF AN ERROR OCCURS, A PRINTOUT EXPLAINING THE NATURE AND LOCATION (BOARD, MODULE, BANK, AND BIT) OF THE ERROR IS GENERATED AND CONTROL IS RETURNED TO DTOS. IF NO ERROR OCCURS, TEST 2 IS RUN. EACH TEST IS RUN IN A SIMILAR MANNER UNTIL AN ERROR OCCURS OR ALL THE TESTS ARE COMPLETED. A COMPLETION MESSAGE IS THEN PRINTED OUT (SEE SECTION III. C) 2.).

B) IN MANUAL MODE

DEFINITION: IN MANUAL MODE, THE OPERATOR CAN TAILOR THE PROGRAM TO EXAMINE MORE SPECIFIC PROBLEMS. THE OPERATOR MUST ENTER THE SETTINGS FOR ALL THE CF'S BEFORE THE TESTS CAN BE RUN (I.E., AT THE START OF THE FIRST PASS). A <LF> RESPONSE WILL SET THE CF TO ITS DEFAULT. SEE TABLE 2 FOR DEFAULT SETTINGS. MANUAL MODE IS RUN BY USING ANY DTOS MANUAL MODE COMMAND (E.G., LOAD MEMD, DEBUG MEMD).

ALL NUMERIC QUESTIONS SHOULD BE ANSWERED IN OCTAL NUMBERS. ALL YES/NO QUESTIONS MUST BE ANSWERED WITH "Y" <CR>, "N" <CR>, OR <LF>. THE <LF> SETS THE CF TO THE DEFAULT, WHICH IS 'NO'.

1. MODULES TO BE TESTED

ENTER A BIT PATTERN TO DETERMINE WHAT MODULES ARE TO BE TESTED. A "0" RESPONSE TESTS ALL MODULES. THIS CF APPLIES TO ALL TESTS EXCEPT THE MODIFIED DATA=ADDRESS AND DATA=ADDRESS UPPER.

MUT	MODULE TESTED	(MUT=MODULES UNDER TEST)
15	A	
14	B	
13	C	
12	D	

AS USUAL, A BIT SET MEANS THAT THAT MODULE WILL BE TESTED.

2. ERROR CONTROL MODE (ECM)

"ECM (0-4)? " -THE NUMBER ENTERED WILL AFFECT THE ACTION TAKEN UPON HITTING AN ERROR:

ECM	ACTION TAKEN
0	PRINT AN ERROR REPORT AND RETURN TO DTOS.
1	PRINT AN ERROR REPORT AND CONTINUE.
2	PRINT AN ERROR REPORT AND HALT.
3	PRINT AN ERROR REPORT AND GO TO PROGRAM MONITOR.
4	ENTER ERROR IN ERROR LOG AND CONTINUE.

ECM=2 IS ILLEGAL WHEN USING POWER SUPPLY VOLTAGE MARGINING. (SEE B) 5.)

THE DEFAULT ECM IS 3.
IF "4" IS ENTERED, THE USER MUST THEN SET THE LOG PRINT (LP) CONTROL FLAG:
"PRINT ERROR LOG AT CONCLUSION (Y/N)? "

FOR FURTHER INFORMATION -- III.B)7., III.C)1., III.D) .

3. SUPPLEMENTAL ERROR INFORMATION

"SUPPLEMENTAL ERROR INFORMATION (Y/N)? "

IF YES, THEN THE FAILURE ADDRESS, EXPECTED DATA, AND ACTUAL DATA ARE PRINTED IN ADDITION TO THE (DEFAULT) ERROR REPORT FORMAT. SEE III.C) .

4. POWER SUPPLY VOLTAGES (PSV)

THE PROGRAM CAN EITHER RUN THE TESTS UNDER NOMINAL CONDITIONS OR UNDER MARGINAL CONDITIONS. TO RUN UNDER MARGINAL VOLTAGES, THE BACKPANEL PROGRAMMING PLUG MUST BE IN THE CORRECT POSITION.

THE USER ENTERS AN OCTAL BIT PATTERN AS FOLLOWS:

BIT SET	VBB (-5V)	VDD (+12V)	VCC (+5V)
15	-5.25	10.8	4.5
14	-4.5V	12.6	5.25

IF NO BITS ARE SET, THE TESTS WILL RUN UNDER NOMINAL VOLTAGES (WHICH IS THE DEFAULT CONDITION). OTHERWISE, THE TESTS WIL BE RUN UNDER EACH CHOSEN MARGINAL CONDITION.

EXAMPLES (OCTAL NUMBERS): 1= HI/LO/LO ONLY; 2= LO/HI/HI ONLY; 3= FIRST RUN WITH HI/LO/LO, SECOND RUN WITH LO/HI/HI; 0= NOMINAL VOLTAGES ONLY.

PSV MAY NOT BE USED WHEN ECM=2 (HALT ON ERROR REPORT). THIS IS TO ASSURE THAT THE VOLTAGES ARE RETURNED TO NOMINAL LEVELS BEFORE ANY OTHER PROGRAM IS RUN. FOR THIS REASON, THE PROGRAM MONITOR'S "H" (HALT) COMMAND WILL BE REJECTED IF PSV>0.

PROGRAM USING SWITCH PACK), 'BREAK', OR HIT THE FRONT PANEL RESET SWITCH DURING ANY RUN WHERE VOLTAGES ARE BEING MARGINED. IF IT IS VITAL TO STOP THE PROGRAM DURING A PSV RUN, USE THE FOLLOWING SEQUENCE:

1) 'CNTRL-C': PUTS YOU IN THE PROGRAM MONITOR.
2) 'A': ABORTS THE PROGRAM, RETURNS VOLTAGES TO NOMINAL LEVELS, AND RETURNS TO DTOS.
3) 'EXIT': A DTOS COMMAND WHICH HALTS THE MACHINE.

IF YOU RESET PSV=0 USING THE PROGRAM MONITOR DURING A PSV RUN, THE PROGRAM WILL SUBSEQUENTLY ALLOW YOU TO USE THE "H" COMMAND. HOWEVER, THIS IS NOT ADVISABLE BECAUSE THIS WILL NOT RESET THE VOLTAGES TO THEIR NOMINAL VOLTAGES. AS A GENERAL RULE, WHEN USING PSV, AND YOU WANT TO STOP THE PROGRAM, USE THE PROCEDURE OUTLINED ABOVE.

AS USUAL, IF AT ANY TIME AN ERROR IS ENCOUNTERED, THE PROGRAM WILL PROCESS IT ACCORDING TO THE PRESENT ERROR CONTROL MODE.

********** ERCC IS NOT IMPLEMENTED ON NOVA 4 MACHINES **********

5. ERCC ENABLE

"ENABLE ERCC (Y/N)? "

TO CHECK THE ERCC BITS, ANSWER "Y". WITH ERCC ENABLED, THE ERCC LOGIC WILL COPY THE ERCC BITS (MEMIN 16-20) INTO DATA BITS 11-15.

IF ERCC IS ENABLED AND AN ERROR IS REPORTED IN BITS 11-15, THEN THE USER SHOULD TRANSLATE THAT TO MEAN THE ERCC BITS (I.E., BITS 16-20). TO REMIND HIM OF THIS, A WARNING MESSAGE WILL BE PRINTED ALONG WITH THE ERROR PRINTOUT, AND IF LOGGING ERRORS, BIT 0 OF THE TEST NAMES' BIT PATTERN WILL BE SET.

NOTE THAT AN ERCC ERROR IS TREATED JUST LIKE ANY OTHER ERROR WHICH WOULD BE DETECTED BY A TEST PATTERN.

"ERCC ENABLE" IS ONLY IMPLEMENTED AT THE BEGINNING OF THE PROGRAM (I.E., CHANGING THE CF'S VALUE IN THE PROGRAM MONITOR WIL HAVE NO AFFECT UNTIL THE NEXT PASS OF THE PROGRAM).

6. ERROR LOG

THE ERROR LOG CONSISTS OF A LIST OF 3-WORD BLOCKS THAT CONTAIN THE FOLLOWING INFORMATION:

WORD #	CONTENTS
1	BOARD, BANK, MODULE, BIT NUMBER
2	TEST NAMES' BIT PATTERN
3	FAILURE COUNT

THE FAILURE COUNT IS THE TOTAL NUMBER OF TIMES AN ERROR HAS OCCURRED UNDER ALL TESTS RUN FOR THE CHIP WHOSE LOCATION IS SPECIFIED IN WORD 1.

LOGGING AN ERROR ENTAILS THE FOLLOWING ACTIONS:

A/ ERROR'S CHIP LOCATION IN LOG?

B/ IF IT IS IN THE LOG, UPDATE THE TEST NAMES' BIT PATTERN, AND INCREMENT THE FAILURE COUNT.

C/ IF IT IS NOT IN THE LOG, CREATE A NEW NODE WITH ALL PERTINENT INFORMATION ENTERED FOR THAT ERROR. UPDATE LOG.END POINTER.

IF AN ATTEMPT TO CREATE ANOTHER NODE WOULD OVERFLOW THE LOG, THEN A WARNING MESSAGE IS ISSUED, AND THE PROGRAM MONITOR IS ENTERED.

FIGURE 2 IS AN EXAMPLE OF AN ERROR LOG PRINTOUT. SEE THE NOTES UNDER FIGURE 2 FOR MORE INFORMATION.

7. TEST PATTERNS

ENTER A BIT PATTERN TO DETERMINE WHAT TESTS ARE TO BE RUN. SEE TABLE 1 FOR TEST NAMES' BIT POSITIONS. A BIT SET WOULD MEAN THAT THAT TEST WILL BE RUN.

A NOTE ON THE OPTIONALLY AVAILABLE PATTERNS-

GALLOPING PATTERN IS A VERY SLOW BUT COMPREHENSIVE TEST PATTERN WHICH OFFERS STRONG TESTS FOR FAULTY ADDRESS DECODERS, ALL TYPES OF CELL INTERACTIVE COUPLING, SLOW ACCESS TIME, AND CELL FUNCTIONALITY.

C) ERROR REPORTING

UPON HITTING AN ERROR, THE PROGRAM PRINTS OUT 1. 2. IS PRINTED OUT AT THE END OF EACH BOARD. 3. IS PRINTED OUT AT THE END OF THE PROGRAM.

IF THE SEI CF IS SET, 4. IS PRINTED ALONG WITH 1.

1. ERROR REPORT FORMAT

```
TEST NAME: _____
BOARD NUMBER= _____
MODULE= __
BANK=
BIT NUMBER(S)= _ _  _ _ _
```

BE APPENDED TO THE ERROR REPORT FORMAT.

2. NORMAL TERMINATION FORMAT

```
TESTING COMPLETED ...

BOARD NUMBER= ____
ADDRESS START= _____
ADDRESS END= _____
RAM TYPE= ____ K
```

THE 'ADDRESS START' REFERRED TO IN THE NORMAL TERMINATION REPORT IS THE BOARD STARTING ADDRESS. THE 'ADDRESS END' IS THE BOARD ENDING ADDRESS.

3. END PROGRAM FORMAT

(MAKES SENSE, HUM?)

4. SUPPLEMENTAL ERROR INFORMATION

```
ADDRESS= _____
EXPECTED DATA= ____
ACTUAL DATA= _____
```

THE SUPPLEMENTAL ERROR INFORMATION IS PRINTED BETWEEN REPORT FORMAT.

D) PROGRAM MONITOR

IN BOTH MANUAL AND AUTO MODES, AT THE END OF EACH TEST, AT THE OCCURRENCE OF AN ERROR, AND AT THE END OF EACH PASS, THE TTY IS TESTED FOR INPUT. IF A VALID SWITCH PACK COMMAND WAS ENTERED, IT IS THEN EXECUTED. IF NOT, A CHECK FOR 'CNTRL-C' IS PERFORMED. IF 'CNTRL-C' WAS TYPED, A SMALL "PROGRAM MONITOR" IS EXECUTED. A "**?" PROMPT SIGNALS THE USER TO ENTER ONE OF THE VALID COMMANDS LISTED IN TABLE 3. ILLEGAL COMMANDS HAVE NO ILL EFFECTS.

ONCE IN THE PROGRAM MONITOR, SWITCH PACK COMMANDS ARE NOT ACCEPTED. FOR THIS REASON, IN ORDER TO GET A HARD COPY OF THE ERROR LOG PRINTOUT, IT IS NECESSARY TO ENABLE SWREG'S BIT 5 BEFORE ENTERING THE PROGRAM MONITOR (I.E., TYPE '5' BEFORE 'CNTRL-C' AND 'P' OR 'D').

ANY CHANGES MADE IN THE PATS, PSV, OR ECCE CF'S WILL NOT TAKE EFFECT UNTIL THE NEXT BOARD (FOR BOARDS 2 THRU LAST) OR THE NEXT BANK (FOR BOARD 1).

IV. DESIGN HIGHLIGHTS

A) ALL GALLOPING TEST PATTERNS WILL BE RUN USING A MICROCODE ROUTINE TO ACHIEVE THE GREATEST SPEED AND TO TEST THE MEMORY TO ITS FULLEST.

B) EXISTING DLIB ROUTINES THAT ARE APPLICABLE ARE TO BE USED.

C) INTERRUPTS WILL BE ENABLED ONLY FOR DETERMINING BOARD CONFIGURATIONS AND DURING POWER SUPPLY VOLTAGE MARGINING WITHIN THE PROGRAM &EXEC. THEY WILL BE DISABLED THE REST OF THE PROGRAM.

D) IF THE MMPU1 EXISTS, THEN NAMDU2 WILL IMMEDIATELY RETURN TO DTOS. IN THIS CASE, IT IS NECESSARY TO RUN NAMDM.

V. KNOWN ANOMALIES

A) ROW AND BIT COLUMN RIPPLING

1/ A BAD CHIP MAY LOAD DOWN A SIGNAL (E.G., RAS OR CAS) SUFFICIENTLY TO CAUSE SOME OR ALL OF THE OTHER CHIPS IN THE SAME BANK-MODULE (I.E.,ROW) OR BIT COLUMN TO GENERATE ERRORS. IN THE ERROR LOG, THE GENUINE BAD CHIP CAN BE SPOTTED AS THE ONE WITH THE MOST ERRORS (A FAILURE COUNT IN THE THOUSANDS), WHILE THE GOOD CHIPS WOULD HAVE A FAILURE COUNT ONLY THE HUNDREDS. IF THIS OCCURS, REPLACE THE CHIP WITH THE HIGHEST NUMBER OF REPORTED ERRORS AND RERUN THE PROGRAM. IF ALL THE OTHER CHIPS WERE REALLY GOOD, THEN ALL THE RELATED ERRORS SHOULD DISAPPEAR.

2/ YOU RUN THE PROGRAM WITH ERROR REPORTING (AS WOULD BE THE CASE IN AUTO MODE), AN ERROR IS REPORTED, AND YOU REPLACE THE REPORTED CHIP. IF, AFTER RERUNNING THE PROGRAM, YOU STILL GET AN ERROR IN THE SAME ROW AND/OR THE SAME BIT COLUMN, THIS MAY BE THE DREADED 'RIPPLING' EFFECT. INSTEAD OF COMMITING SUICIDE, RETURN THE OFFENDED CHIP (WITH APPROPRIATE APOLOGIES) TO THE BOARD AND RERUN THE PROGRAM USING ERROR LOGGING, FOLLOWING THE PROCEDURE OUTLINED IN '1/'.

3/ IF THE LOADING FROM THE GENUINE BAD CHIP IS SEVERE ENOUGH, IT CAN PULL A CRITICAL SIGNAL DOWN OR UP AND CAUSE ALL THE "GOOD" CHIPS IN THE SAME ROW AND/OR BIT COLUMN TO GENERATE AS MANY ERRORS AS THE BAD CHIP ITSELF. IN THIS CASE, YOU'RE OUT OF LUCK, AND MUST RESORT TO EMULATING A BIRD (IN OTHER WORDS, THE 'OL "HUNT AND PECK" METHOD).

B) IF YOU TYPE 'CNTRL-C' DURING AN ERROR REPORT WHEN ECM=3 (ENTER PROGRAM MONITOR UPON ERROR), WHEN IT COMES TIME TO 'EXIT PROGRAM MONITOR', YOU WILL HAVE TO ENTER THE 'E' COMMAND TWICE-ONCE FOR YOUR 'CNTRL-C' AND ANOTHER BECAUSE ECM=3.

TABLES AND FIGURES FOLLOW

TABLE 1: TEST NAMES' BIT POSITIONS

BIT POSITION	TEST NAME	MODE RUN
15	MODIFIED DATA=ADDRESS	A
14	STUCK ADDRESS BIT	A
13	MARCHING 1/0	A
12	GALLOPING COLUMNS	A
11	GALLOPING ROWS	A
10	GALLOPING PATTERN	M

NOTES: 1/ A=AUTO; M=MANUAL

2/ IT SHOULD TAKE LESS THAN 2 MINUTES TO RUN ALL THE TESTS. THIS ASSUMES NO MEMORY ERRORS ARE DETECTED (I.E., IF THERE ARE ERRORS, AND ECM=4 (ERRORS BEING LOGGED, NOT PRINTED), THEN THE TESTS WILL TAKE LONGER TO COMPLETE).

10013 NMDU2

TABLE 2: CONTROL FLAGS' DEFAULT SETTINGS

DESCRIPTION OF CF	CF	PROGRAM OCTAL VALUE	DEFAULT
MODULES UNDER TEST	MUT	0	ALL
ERROR CONTROL MODE	ECM	0	PRINT ERROR, RETURN TO DTO
SUPPL. ERROR INFO.	SEI	0	NONE.
POWER SUPPLY VOLTAGES	PSV	0	NOMINAL ONLY
ECC ENABLE	ECCE	0	NOT AVAIL./NOT ENABLED
ERROR LOG PRINT	LP	0	NO LOG PRINT @ CONCL.
PATTERNS TO BE RUN	PATS	37	ALL EXCEPT GALPAT

NOTES: 1/ CF=CONTROL FLAG

2/ THE DEFAULT PATTERNS ARE THE FIRST 5 PATTERNS LISTED LISTED IN TABLE 1.

3/ 'PROGRAM OCTAL VALUE' REFERS TO THE VALUE THE 'CF' WOULD CONTAIN TO IMPLEMENT THE DESCRIBED DEFAULT.

10014 NMDU2

TABLE 3: PROGRAM MONITOR COMMANDS

COMMAND	(STANDS FOR)	ACTION TAKEN
A	ABORT	RETURN TO DTOS
C	CLEAR	CLEAR ERROR LOG
D	DUMP	PRINT & CLEAR ERROR LOG
E	EXIT	RETURN TO MAIN PROGRAM
F	FLAGS	PRINT CONTROL FLAGS
H	HALT	PROGRAM HALTED
P	PRINT	PRINT ERROR LOG
R	RESET	PRINT CF'S; INPUT NEW VALUES
T	TERMINATE	TERMINATE THE CURRENT TEST, AND RETURN TO THE PROGRAM EXEC.

NOTES: 1/ AN ILLEGAL COMMAND WILL CAUSE A MESSAGE TO BE PRINTED, LISTING THE VALID COMMANDS THAT MAY BE ENTERED.

10015 NMDU2

FIGURE 1: FAULT TYPES VS. MEMORY TEST PATTERNS

TEST NAME	MAJOR FAULT TYPES TESTED
MODIFIED D=A	1,2,8
STUCK ADR. BIT	1,2
MARCHING 1/0	2,8
GALLOPING COL.	4,5
GALLOPING ROWS	3,5,7
GALLOP'G PATTERN	1,2,3,4,6,7,8

KEY:

1: FAULTY ADDRESS DECODING- WRONG CELL OR CHIP ACCESSED
2: FAULTY ADDRESS DECODING- CELL INACCESSIBLE
3: INTERACTIVE COUPLING WITHIN ROWS
4: INTERACTIVE COUPLING WITHIN COLUMNS
5: REFRESH SENSITIVITY
6: SLOW ACCESS TIME
7: SENSE AMPLIFIER RECOVERY
8: CELL FUNCTIONALITY

10016 NMDU2

FIGURE 2: ERROR LOG OUTPUT

BOARD	CHIP LOCATION BANK	MOD	BIT	TEST NAMES	FAILURE COUNT
1	1	C	9	000140	360
1	1	C	12	000140	360
1	1	D	0	000140	32157
1	0	A	6	000202	3
2	0	B	9	000040	64
2	0	B	10	000040	64
2	0	B	15	000040	64

NOTES:

1/ 'TEST NAME(S)' IS A BIT PATTERN. A BIT SET MEANS THAT AN ERROR HAS OCCURRED UNDER THE CORRESPONDING TEST LISTED IN TABLE 1 FOR THE SPECIFIED CHIP LOCATION. (E.G., 41= BITS 10,15 => GALLOPING ROWS AND MODIFIED DATA=ADDRESS, IF RUNNING THE MAPPED VERSION). BIT 0 SET MEANS THAT ERCC WAS ENABLED DURING TESTING.

2/ THE TEST NAMES LISTED INDICATE UNDER WHAT TESTS AN ERROR HAS OCCURRED IN THE WORD.

3/ 'FAILURE COUNT' IS THE NUMBER OF TIMES AN ERROR HAS OCCURRED AT THE SPECIFIED CHIP LOCATION.
IF THE FAILURE COUNT=177777, THEN THERE WAS AN OVERFLOW IN THE FAILURE COUNT DATA WORD (I.E., THE NUMBER OF ERRORS FOR THAT CHIP WAS >=177777).

4/ ALL NUMBERS ARE IN OCTAL.

5/ TO GET A HARD COPY OF THE ERROR LOG PRINTOUT, IT IS NECESSARY TO SET SWREG'S BIT 5 BEFORE ENTERING THE PROGRAM MONITOR (UNLESS, OF COURSE, THE CONSOLE DEVICE IS A PRINTER OR TELETYPE). IF THE LOG IS TO BE PRINTED OUT ONLY AT THE CONCLUSION OF THE PROGRAM, SET BIT 5 ANYTIME DURING THE EXECUTION OF THE PROGRAM.

6/ A 'GALLOPING PATTERN' FAILURE COUNT OF 1 SHOULD BE INTERPRETED AS A HARD ERROR. ("1" IS THE MAXIMUM FAILURE COUNT THAT GALPAT WILL GENERATE).

```
10017  NMDU2
01
02                          FIGURE 31   MEMORY TOPOLOGY, 4K RAMS
03                          ========
04
05          BIT 15          ..... ..... ..... ..... ..... ..... .....      BIT 0
06
07          -----------------                                              ---------
08          !0 400 ... 374001!                                             !       !
09          !4             !!              BANK 0                          !       !
10          !.             !!                                              !       !
11          !.             !!                                              !       !
12          !374       377741!                                             !       !
13          -----------------                                              ---------
14          -----------------                                              ---------
15          !40000...  774001!                                             !       !
16          !40004         !!                                              !       !
17          !.             !!              BANK 1                          !       !
18          !40374...  777741!                                             !       !
19          -----------------                                              ---------
20
21          ================================================================== **
22
23          -----------------                                              ---------
24          !1 401 ... 374011!                                             !       !
25          !5             !!              BANK 0                          !       !
26          !.             !!                                              !       !
27          !.             !!                                              !       !
28          !375       377751!                                             !       !
29          -----------------                                              ---------
30          -----------------                                              -------- .
31          !40001...  774011!                                             !       !
32          !40005         !!                                              !       !
33          !.             !!              BANK 1                          !       !
34          !40375...  777751!                                             !       !
35          -----------------                                              ---------
36
37          ================================================================================
38
39          -----------------                                              -------- .
40          !2 402 ... 374021!                                             !       !
41          !6             !!              BANK 0                          !       !
42          !.             !!                                              !       !
43          !.             !!                                              !       !
44          !376       377761!                                             !       !
45          -----------------                                              -------- .
46          -----------------                                              ---------
47          !40002...  774021!                                             !       !
48          !40006         !!                                              !       !
49          !.             !!              BANK 1                          !       !
50          !40376...  777761!                                             !       !
51          -----------------                                              -------- .
52
53          ================================================================================
```

```
10018  NMDU2
01          -----------------                                              ---------
02          !3 403 ... 374031!                                             !       !
03          !7             !!              BANK 0                          !       !
04          !.             !!                                              !       !
05          !.             !!                                              !       !
06          !377       377771!                                             !       !
07          -----------------                                              ---------
08          -----------------                                              ---------
09          !40003...  774031!                                             !       !
10          !40007         !!                                              !       !
11          !.             !!              BANK 1                          !       !
12          !40377...  777771!                                             !       !
13          -----------------                                              ---------
14
15
16                  EACH BLOCK REPRESENTS A CHIP. THE NUMBERS WITHIN THE BLOCK DENOTE
17          PHYSICAL ADDRESSES.
18                  THIS IS THE VIEW IF THE FRONT END (I.E., THE ONE WITH THE PINS)
19          IS TOWARDS YOU. THE BANKS ALTERNATE; THE MODULES RUN BACK TO FRONT (A-D);
20          THE BIT NUMBERS RUN RIGHT TO LEFT. IF THERE ARE ANY ERCC CHIPS, THEY WOULD
21          BE APPENDED TO THE LEFT.
22                  A 16K BOARD WOULD HAVE ONLY BANK 0. A 32K BOARD WOULD HAVE BOTH
23          BANK 0 AND BANK 1.
24
25
26                  .ENDC
27                  .EJEC
```

```
 0001  NMDU1     AOS ASSEMBLER REV 02.02         14:42:50  12/28/78
01
02
03
04
05
06
07              ;*****************************************************************
08              ;
09              ; NAME: N4MDU1.TX                          PART NUMBER: 097-001134
10              ;
11              ;
12              ; DESCRIPTION: NOVA 4 MEMORY DIAGNOSTIC / UNMAPPED / PART 1
13              ;
14              ;
15              ; REVISION HISTORY:
16              ;
17              ;       REV.            DATE
18              ;
19              ;       00              12/27/78
20              ;
21              ;
22              ; COPYRIGHT (C) DATA GENERAL CORPORATION, 1978
23              ; ALL RIGHTS RESERVED.
24              ;*****************************************************************
```

N4MDU

```
 0002  NMDU1
01      000000          .DO     0
02
03                                      CONTENTS
04                                      ========
05
06
07              I. GOAL
08
09              II. DESIRED ATTRIBUTES
10
11              III. WHAT THE PROGRAM WILL DO AND WHY
12
13              IV. DESIGN HIGHLIGHTS
14
15              V. KNOWN ANOMALIES
16
17
18              TABLE 1- TEST NAMES' BIT POSITIONS
19
20              TABLE 2- CONTROL FLAGS' DEFAULT SETTINGS
21
22              TABLE 3- PROGRAM MONITOR COMMANDS
23
24
25              FIGURE 1- FAULT TYPES VS. MEMORY TEST PATTERNS
26
27              FIGURE 2- ERROR LOG OUTPUT
28
29              FIGURE 3- MEMORY TOPOLOGY, 4K RAMS
30
31
32                      LAST MODIFICATION:      12/27/78    14:42
```

I. GOAL

PROVIDE A MEANS OF TESTING UP TO 32K WORDS OF MOS DYNAMIC RAMS ON AN UNMAPPED NOVA 4.

II. DESIRED ATTRIBUTES

A) SPEED- ENTIRE PROGRAM SHOULD RUN NO LONGER THAN 10-15 MINUTES IN AUTO MODE.

B) THOROUGHNESS- ALL MEMORY FAULT TYPES SHOULD BE TESTED.

C) STRENGTH- EACH MEMORY FAULT TYPE SHOULD BE CHECK BY A MEMORY TEST PATTERN OF AT LEAST MEDIUM STRENGTH. THE STRONGER A TEST, THE MORE LIKELY IT IS TO FIND A BUG. IN A SIMILAR VEIN, "WEAK" TESTS FIND HARD ERRORS, "STRONG" TESTS FIND SOFT (I.E., INTERMITTENT, INTERACTIVE) ERRORS.

D) DTOS COMPATIBLE.

E) BE USED UNDER ANY OF THE FOLLOWING POSSIBLE CONFIGURATIONS:

CONFIG=0	(1) 16K BOARD
CONFIG=1	(1) 32K BOARD
CONFIG=2	(2) 16K BOARDS

F) OPTIONS- UNDER MANUAL CONTROL, THE USER MAY OPT FOR TESTS WHICH COVER LESS IMPORTANT FAULT TYPES AND/OR TAKE A LONG TIME TO RUN. HE CAN ALSO ENABLE SPECIAL FEATURES WHICH AID IN THE LOCATION OF DIFFICULT ERRORS.

G) IF AN ERROR IS FOUND, PRINT OUT THE EXACT LOCATION OF THE BAD CHIP.

H) PROGRAM LENGTH WILL BE 8K OR LESS. THIS ALLOWS THE TESTING OF 4K RAMS/16K BOARD SYSTEMS.

I) AN ERROR LOG BE KEPT, WHICH CONTAINS A HISTORY OF ALL ERRORS ENCOUNTERED.

III. WHAT THE PROGRAM WILL DO AND WHY

A) IN AUTO MODE

DEFINITION: IN AUTO MODE, ALL CONTROL FLAGS (CF'S) ARE SET TO DEFAULT. SEE TABLE 2 FOR DEFAULT SETTINGS. 'AUTO MODE' IS RUN BY ISSUING ANY AUTO MODE COMMAND IN DTOS (E.G., LOAD.NN MEMD, AUTO). ALSO, AUTO MODE CAN BE SIMULATED BY RUNNING UNDER DTOS MANUAL MODE BUT USING THE DEFAULT SETTINGS FOR ALL CF'S.

02
03 1. SIZE MEMORY- RUN A TOP DOWN MEMORY SIZING ROUTINE TO
04 DETERMINE THE TOP OF MEMORY.
05 DETERMINE THE CONFIGURATION OF THE MACHINE. THE
06 THREE POSSIBLE BOARD CONFIGURATIONS ARE LISTED IN II. E).
07 THIS IS NECESSARY FOR CONTROL OF VARIOUS PARAMETERS WITHIN THE
08 PROGRAM, ESPECIALLY IN DETERMINING THE EXACT LOCATION OF ANY
09 ERRORS (BOARD, MODULE, BANK, BIT).
10
11 2. RUN THE SPECIFIED TESTS
12
13 SIX TESTS HAVE BEEN CHOSEN THAT MOST OPTIMALLY MEET
14 THE DESIRED ATTRIBUTES: 1) MODIFIED DATA=ADDRESS; 2) STUCK
15 ADDRESS BIT; 3) MARCHING 1/0; 4) GALLOPING COLUMN; 5) GALLOPING
16 ROWS; AND 6) GALLOPING DIAGONAL.
17
18 THE TESTS WILL BE RUN IN THAT ORDER. THIS WAY, IF THE
19 FAULT IS A HARD ERROR, THEN ONE THE FIRST THREE QUICK TESTS
20 WILL FIND IT. IF NOT, THE GALLOPING TESTS WILL THEN BE RUN.
21 SEE TABLE 1 FOR APPROXIMATE TEST TIMES FOR EACH PATTERN.
22
23 THE MODIFIED DATA=ADDRESS TEST WILL BE RUN IN TWO
24 PHASES. IN THE FIRST PHASE OF THE MODIFIED DATA=ADDRESS TEST,
25 THE DATA WORD WRITTEN TO AND SUBSEQUENTLY READ FROM THE MEMORY
26 LOCATION WILL BE THE LOWER 16 BITS OF ITS 20-BIT ADDRESS. ON THE
27 SECOND PHASE, THE COMPLEMENT OF THE LOWER 16 BITS OF THE 20-BIT
28 ADDRESS WILL BE WRITTEN TO/READ FROM THE LOCATIONS.
29
30 THE GALLOPING TEST PATTERNS WILL BE RUN IN 16K CHUNKS.
31 EXTENSIVE USE OF THE MMPU1 FEATURE AND MICROCODE SUPPORT IS MADE.
32
33 THESE SIX TESTS COVER THE FOLLOWING SIX FAULT TYPES:
34 1) FAULTY ADDRESS DECODERS; 2) INTERACTIVE COUPLING BETWEEN
35 CELLS; 3) REFRESH SENSITIVITY; 4) SLOW ACCESS TIME; 5) SENSE
36 AMPLIFIER RECOVERY; 6) CELL FUNCTIONALITY (CAN THE CELL HOLD
37 WHAT IT'S TOLD TO); AND 7) STUCK ADDRESS BITS AT THE CHIP LEVEL.
38
39 FIGURE 1 DETAILS WHICH TESTS COVER WHAT FAULT TYPES.
40
41 EACH TEST WILL BE RUN SEQUENTIALLY DEPENDENT ON ERROR
42 CONDITIONS. SPECIFICALLY, TEST 1 IS RUN. IF AN ERROR OCCURS, A
43 PRINTOUT EXPLAINING THE NATURE AND LOCATION (BOARD, MODULE,
44 BANK, AND BIT) OF THE ERROR IS GENERATED AND CONTROL IS RETURNED
45 TO DTOS. IF NO ERROR OCCURS, TEST 2 IS RUN. EACH TEST IS RUN
46 IN A SIMILAR MANNER UNTIL AN ERROR OCCURS OR ALL THE TESTS
47 ARE COMPLETED. A COMPLETION MESSAGE IS THEN PRINTED OUT (SEE
48 SECTION III. C) 2.).

02
03
04
05 B) IN MANUAL MODE
06 ---
07
08 DEFINITION: IN MANUAL MODE, THE OPERATOR CAN TAILOR THE PROGRAM
09 TO EXAMINE MORE SPECIFIC PROBLEMS. THE OPERATOR MUST ENTER
10 THE SETTINGS FOR ALL THE CF'S BEFORE THE TESTS CAN BE RUN
11 (I.E., AT THE START OF THE FIRST PASS). A <LF> RESPONSE WILL SET THE
12 CF TO ITS DEFAULT. SEE TABLE 2 FOR DEFAULT SETTINGS. MANUAL MODE
13 IS RUN BY USING ANY DTOS MANUAL MODE COMMAND (E.G., LOAD MEMD,
14 DEBUG MEMD).
15
16 ALL NUMERIC QUESTIONS SHOULD BE ANSWERED IN OCTAL
17 NUMBERS. ALL YES/NO QUESTIONS MUST BE ANSWERED WITH "Y" <CR>,
18 "N" <CR>, OR <LF>. THE <LF> SETS THE CF TO THE DEFAULT, WHICH
19 IS 'NO'.
20
21
22 1. MODULES TO BE TESTED
23
24 ENTER A BIT PATTERN TO DETERMINE WHAT MODULES ARE
25 TO BE TESTED. A "0" RESPONSE TESTS ALL MODULES. THIS CF
26 APPLIES TO ALL TESTS EXCEPT THE MODIFIED DATA=ADDRESS AND
27 DATA=ADDRESS UPPER.
28
29 | MUT | MODULE TESTED | (MUT=MODULES UNDER TEST) |
30 |---|---|---|
31 | 15 | A | |
32 | 14 | B | |
33 | 13 | C | |
34 | 12 | D | |
35
36 AS USUAL, A BIT SET MEANS THAT THAT MODULE WILL BE
37 TESTED.
38
39 2. ERROR CONTROL MODE (ECM)
40
41 "ECM (0-4)? " -THE NUMBER ENTERED WILL AFFECT THE
42 ACTION TAKEN UPON HITTING AN ERROR:
43
44 | ECM | ACTION TAKEN |
45 |---|---|
46 | 0 | PRINT AN ERROR REPORT AND RETURN TO DTOS. |
47 | 1 | PRINT AN ERROR REPORT AND CONTINUE. |
48 | 2 | PRINT AN ERROR REPORT AND HALT. |
49 | 3 | PRINT AN ERROR REPORT AND GO TO PROGRAM MONITOR. |
50 | 4 | ENTER ERROR IN ERROR LOG AND CONTINUE. |
51
52 ECM=2 IS ILLEGAL WHEN USING POWER SUPPLY VOLTAGE
53 MARGINING. (SEE B) 5.)
54
55 THE DEFAULT ECM IS 3.
56 IF "4" IS ENTERED, THE USER MUST THEN SET THE LOG
57 PRINT (LP) CONTROL FLAG:
58 "PRINT ERROR LOG AT CONCLUSION (Y/N)? "
59
60 FOR FURTHER INFORMATION -- III.B)7., III.C)1., III.D) .

3. SUPPLEMENTAL ERROR INFORMATION

"SUPPLEMENTAL ERROR INFORMATION (Y/N)? "

IF YES, THEN THE FAILURE ADDRESS, EXPECTED DATA, AND ACTUAL DATA ARE PRINTED IN ADDITION TO THE (DEFAULT) ERROR REPORT FORMAT. SEE III.C) .

4. POWER SUPPLY VOLTAGES (PSV)

THE PROGRAM CAN EITHER RUN THE TESTS UNDER NOMINAL CONDITIONS OR UNDER MARGINAL CONDITIONS. TO RUN UNDER MARGINAL VOLTAGES, THE BACKPANEL PROGRAMMING PLUG MUST BE IN THE CORRECT POSITION.

THE USER ENTERS AN OCTAL BIT PATTERN AS FOLLOWS:

BIT SET	VBB (-5V)	VDD (+12V)	VCC (+5V)
15	-5.25	10.8	4.5
14	-4.5	12.6	5.25

IF NO BITS ARE SET, THE TESTS WILL RUN UNDER NOMINAL VOLTAGES (WHICH IS THE DEFAULT CONDITION). OTHERWISE, THE TESTS WIL BE RUN UNDER EACH CHOSEN MARGINAL CONDITION.

EXAMPLES (OCTAL NUMBERS): 1= HI/LO/LO ONLY; 2= LO/HI/HI ONLY; 3= FIRST RUN WITH HI/LO/LO, SECOND RUN WITH LO/HI/HI; 0= NOMINAL VOLTAGES ONLY.

PSV MAY NOT BE USED WHEN ECM=2 (HALT ON ERROR REPORT). THIS IS TO ASSURE THAT THE VOLTAGES ARE RETURNED TO NOMINAL LEVELS BEFORE ANY OTHER PROGRAM IS RUN. FOR THIS REASON, THE PROGRAM MONITOR'S "H" (HALT) COMMAND WILL BE REJECTED IF PSV>0.

PROGRAM USING SWITCH PACK), 'BREAK', OR HIT THE FRONT PANEL RESET SWITCH DURING ANY RUN WHERE VOLTAGES ARE BEING MARGINED. IF IT IS VITAL TO STOP THE PROGRAM DURING A PSV RUN, USE THE FOLLOWING SEQUENCE:

1) 'CNTRL-C': PUTS YOU IN THE PROGRAM MONITOR.
2) 'A': ABORTS THE PROGRAM, RETURNS VOLTAGES TO NOMINAL LEVELS, AND RETURNS TO DTOS.
3) 'EXIT': A DTOS COMMAND WHICH HALTS THE MACHINE.

IF YOU RESET PSV=0 USING THE PROGRAM MONITOR DURING A PSV RUN, THE PROGRAM WILL SUBSEQUENTLY ALLOW YOU TO USE THE "H" COMMAND. HOWEVER, THIS IS NOT ADVISABLE BECAUSE THIS WILL NOT RESET THE VOLTAGES TO THEIR NOMINAL VOLTAGES. AS A GENERAL RULE, WHEN USING PSV, AND YOU WANT TO STOP THE PROGRAM, USE THE PROCEDURE OUTLINED ABOVE.

AS USUAL, IF AT ANY TIME AN ERROR IS ENCOUNTERED, THE PROGRAM WILL PROCESS IT ACCORDING TO THE PRESENT ERROR CONTROL MODE.

5. ERCC ENABLE

"ENABLE ERCC (Y/N)? "

TO CHECK THE ERCC BITS, ANSWER "Y". WITH ERCC ENABLED, THE ERCC LOGIC WILL COPY THE ERCC BITS (MEMIN 16-20) INTO DATA BITS 11-15.

IF ERCC IS ENABLED AND AN ERROR IS REPORTED IN BITS 11-15, THEN THE USER SHOULD TRANSLATE THAT TO MEAN THE ERCC BITS (I.E., BITS 16-20). TO REMIND HIM OF THIS, A WARNING MESSAGE WILL BE PRINTED ALONG WITH THE ERROR PRINTOUT, AND IF LOGGING ERRORS, BIT 0 OF THE TEST NAMES' BIT PATTERN WILL BE SET.

NOTE THAT AN ERCC ERROR IS TREATED JUST LIKE ANY OTHER ERROR WHICH WOULD BE DETECTED BY A TEST PATTERN.

"ERCC ENABLE" IS ONLY IMPLEMENTED AT THE BEGINNING OF THE PROGRAM (I.E., CHANGING THE CF'S VALUE IN THE PROGRAM MONITOR WIL HAVE NO AFFECT UNTIL THE NEXT PASS OF THE PROGRAM).

6. ERROR LOG

THE ERROR LOG CONSISTS OF A LIST OF 3-WORD BLOCKS THAT CONTAIN THE FOLLOWING INFORMATION:

WORD #	CONTENTS
1	BOARD, BANK, MODULE, BIT NUMBER
2	TEST NAMES' BIT PATTERN
3	FAILURE COUNT

THE FAILURE COUNT IS THE TOTAL NUMBER OF TIMES AN ERROR HAS OCCURRED UNDER ALL TESTS RUN FOR THE CHIP WHOSE LOCATION IS SPECIFIED IN WORD 1.

LOGGING AN ERROR ENTAILS THE FOLLOWING ACTIONS:

A/ ERROR'S CHIP LOCATION IN LOG?

B/ IF IT IS IN THE LOG, UPDATE THE TEST NAMES' BIT PATTERN, AND INCREMENT THE FAILURE COUNT.

C/ IF IT IS NOT IN THE LOG, CREATE A NEW NODE WITH ALL PERTINENT INFORMATION ENTERED FOR THAT ERROR, UPDATE LOG.END POINTER. '

IF AN ATTEMPT TO CREATE ANOTHER NODE WOULD OVERFLOW THE LOG, THEN A WARNING MESSAGE IS ISSUED, AND THE PROGRAM MONITOR IS ENTERED.

FIGURE 2 IS AN EXAMPLE OF AN ERROR LOG PRINTOUT. SEE THE NOTES UNDER FIGURE 2 FOR MORE INFORMATION.

7. TEST PATTERNS

ENTER A BIT PATTERN TO DETERMINE WHAT TESTS ARE TO BE RUN. SEE TABLE 1 FOR TEST NAMES' BIT POSITIONS. A BIT SET WOULD MEAN THAT THAT TEST WILL BE RUN.

A NOTE ON THE OPTIONALLY AVAILABLE PATTERNS-

GALLOPING PATTERN IS A VERY SLOW BUT COMPREHENSIVE TEST PATTERN WHICH OFFERS STRONG TESTS FOR FAULTY ADDRESS DECODERS, ALL TYPES OF CELL INTERACTIVE COUPLING, SLOW ACCESS TIME, AND CELL FUNCTIONALITY.

C) ERROR REPORTING

UPON HITTING AN ERROR, THE PROGRAM PRINTS OUT 1. 2. IS PRINTED OUT AT THE END OF EACH BOARD. 3. IS PRINTED OUT AT THE END OF THE PROGRAM.

IF THE SEI CF IS SET, 4. IS PRINTED ALONG WITH 1.

1. ERROR REPORT FORMAT

```
TEST NAME: _____
BOARD NUMBER= _____
MODULE= __
BANK= __
BIT NUMBER(S)= _ _ _ _ _ _
```

BE APPENDED TO THE ERROR REPORT FORMAT.

2. NORMAL TERMINATION FORMAT

```
TESTING COMPLETED ...

BOARD NUMBER= _____
ADDRESS START= _____
ADDRESS END= _____
RAM TYPE= '___ K
```

THE 'ADDRESS START' REFERRED TO IN THE NORMAL TERMINATION REPORT IS THE BOARD STARTING ADDRESS. THE 'ADDRESS END' IS THE BOARD ENDING ADDRESS.

3. END PROGRAM FORMAT

(MAKES SENSE, HUH?)

4. SUPPLEMENTAL ERROR INFORMATION

```
ADDRESS= _____
EXPECTED DATA= _____
ACTUAL DATA= _____
```

THE SUPPLEMENTAL ERROR INFORMATION IS PRINTED BETWEEN REPORT FORMAT.

D) PROGRAM MONITOR

IN BOTH MANUAL AND AUTO MODES, AT THE END OF EACH TEST, AT THE OCCURRENCE OF AN ERROR, AND AT THE END OF EACH PASS, THE TTY IS TESTED FOR INPUT. IF A VALID SWITCH PACK COMMAND WAS ENTERED, IT IS THEN EXECUTED. IF NOT, A CHECK FOR 'CNTRL-C' IS PERFORMED. IF 'CNTRL-C' WAS TYPED, A SMALL "PROGRAM MONITOR" IS EXECUTED. A "**?" PROMPT SIGNALS THE USER TO ENTER ONE OF THE VALID COMMANDS LISTED IN TABLE 3. ILLEGAL COMMANDS HAVE NO ILL EFFECTS.

ONCE IN THE PROGRAM MONITOR, SWITCH PACK COMMANDS ARE NOT ACCEPTED. FOR THIS REASON, IN ORDER TO GET A HARD COPY OF THE ERROR LOG PRINTOUT, IT IS NECESSARY TO ENABLE SWREG'S BIT 5 BEFORE ENTERING THE PROGRAM MONITOR (I.E., TYPE '5' BEFORE 'CNTRL-C' AND 'P' OR 'D').

ANY CHANGES MADE IN THE PATS, PSV, OR ECCE CF'S WILL NOT TAKE EFFECT UNTIL THE NEXT BOARD (FOR BOARDS 2 THRU LAST) OR THE NEXT BANK (FOR BOARD 1).

IV. DESIGN HIGHLIGHTS

A) ALL GALLOPING TEST PATTERNS WILL BE RUN USING A MICROCODE ROUTINE TO ACHIEVE THE GREATEST SPEED AND TO TEST THE MEMORY TO ITS FULLEST.

B) MEMORY SIZING WILL TAKE ADVANTAGE OF A MICROCODE ROUTINE THAT ACCESSES TWO MEMORY LOCATIONS FAST ENOUGH TO CAUSE MEMORY INTERFERENCE. USE OF THE REAL TIME CLOCK WILL COMPLETE THE MEMORY SIZING ALGORITHM.

C) EXISTING DLIB ROUTINES THAT ARE APPLICABLE ARE TO BE USED.

D) INTERRUPTS WILL BE ENABLED ONLY FOR DETERMINING BOARD CONFIGURATIONS AND DURING POWER SUPPLY VOLTAGE MARGINING WITHIN THE PROGRAM EXEC. THEY WILL BE DISABLED THE REST OF THE PROGRAM.

E) IF THE MMPU1 EXISTS, THEN NMDU1 WILL IMMEDIATELY RETURN TO DTOS. IN THIS CASE, IT IS NECESSARY TO RUN NMDM.

V. KNOWN ANOMALIES

A) ROW AND BIT COLUMN RIPPLING

1/ A BAD CHIP MAY LOAD DOWN A SIGNAL (E.G., RAS OR CAS) SUFFICIENTLY TO CAUSE SOME OR ALL OF THE OTHER CHIPS IN THE SAME BANK-MODULE (I.E.,ROW) OR BIT COLUMN TO GENERATE ERRORS. IN THE ERROR LOG, THE GENUINE BAD CHIP CAN BE SPOTTED AS THE ONE WITH THE MOST ERRORS (A FAILURE COUNT IN THE THOUSANDS), WHILE THE GOOD CHIPS WOULD HAVE A FAILURE COUNT ONLY THE HUNDREDS. IF THIS OCCURS, REPLACE THE CHIP WITH THE HIGHEST NUMBER OF REPORTED ERRORS AND RERUN THE PROGRAM. IF ALL THE OTHER CHIPS WERE REALLY GOOD, THEN ALL THE RELATED ERRORS SHOULD DISAPPEAR.

2/ YOU RUN THE PROGRAM WITH ERROR REPORTING (AS WOULD BE THE CASE IN AUTO MODE). AN ERROR IS REPORTED, AND YOU REPLACE THE REPORTED CHIP. IF, AFTER RERUNNING THE PROGRAM, YOU STILL GET AN ERROR IN THE SAME ROW AND/OR THE SAME BIT COLUMN, THIS MAY BE THE DREADED 'RIPPLIN EFFECT. INSTEAD OF COMMITING SUICIDE, RETURN THE OFFENDED CHIP (WITH APPROPRIATE APOLOGIES) TO THE BOARD AND RERUN THE PROGRAM USING ERROR LOGGING, FOLLOWING THE PROCEDURE OUTLINED IN '1/'.

3/ IF THE LOADING FROM THE GENUINE BAD CHIP IS SEVERE ENOUGH, IT CAN PULL A CRITICAL SIGNAL DOWN OR UP AND CAUSE ALL THE "GOOD" CHIPS IN THE SAME ROW AND/OR BIT COLUMN TO GENERATE AS MANY ERRORS AS THE BAD CHIP ITSELF. IN THIS CASE, YOU'RE OUT OF LUCK, AND MUST RESORT TO EMULATING A BIRD (IN OTHER WORDS, THE 'OL "HUNT AND PECK" METHOD).

B) IF YOU TYPE 'CNTRL-C' DURING AN ERROR REPORT WHEN ECM=3 (ENTER PROGRAM MONITOR UPON ERROR), WHEN IT COMES TIME TO 'EXIT PROGRAM MONITOR', YOU WILL HAVE TO ENTER THE 'E' COMMAND TWICE- ONCE FOR YOUR 'CNTRL-C' AND ANOTHER BECAUSE ECM=3.

TABLES AND FIGURES FOLLOW

TABLE 1: TEST NAMES' BIT POSITIONS

BIT POSITION	TEST NAME	MODE RUN	ESTIMATED TEST TIME
15	MODIFIED DATA=ADDRESS	A	5 SEC
14	STUCK ADDRESS BIT	A	1 SEC
13	MARCHING 1/0	A	5 SEC
12	GALLOPING COLUMNS	A	15 SEC
11	GALLOPING ROWS	A	15 SEC
10	GALLOPING DIAGONAL	A	15 SEC
9	GALLOPING PATTERN	M	1 MIN

NOTES: 1/ A=AUTO; M=MANUAL

2/ ALL ESTIMATED TEST TIMES ARE FOR 32K WORDS OF MEMORY.

3/ TEST TIMES LISTED ASSUME NO MEMORY ERRORS ARE DETECTED (I.E., IF THERE ARE ERRORS, AND ECM=4 (ERRORS BEING LOGGED, NOT PRINTED), THEN THE TESTS WILL TAKE LONGER TO COMPLETE).

TABLE 2: CONTROL FLAGS' DEFAULT SETTINGS

DESCRIPTION OF CF	CF	PROGRAM OCTAL VALUE	DEFAULT
MODULES UNDER TEST	MUT	0	ALL
ERROR CONTROL MODE	ECM	0	PRINT ERROR, RETURN TO DTO
SUPPL. ERROR INFO.	SEI	0	NONE.
POWER SUPPLY VOLTAGES	PSV	0	NOMINAL ONLY
ECC ENABLE	ECCE	0	NOT AVAIL./NOT ENABLED
ERROR LOG PRINT	LP	0	NO LOG PRINT @ CONCL.
PATTERNS TO BE RUN	PATS	77	ALL EXCEPT GALPAT

NOTES: 1/ CF=CONTROL FLAG

2/ THE DEFAULT PATTERNS ARE THE FIRST 6 PATTERNS LISTED IN TABLE 1.

3/ 'PROGRAM OCTAL VALUE' REFERS TO THE VALUE THE 'CF' WOULD CONTAIN TO IMPLEMENT THE DESCRIBED DEFAULT.

TABLE 3: PROGRAM MONITOR COMMANDS

COMMAND	(STANDS FOR)	ACTION TAKEN
A	ABORT	RETURN TO DTOS
C	CLEAR	CLEAR ERROR LOG
D	DUMP	PRINT & CLEAR ERROR LOG
E	EXIT	RETURN TO MAIN PROGRAM
F	FLAGS	PRINT CONTROL FLAGS
H	HALT	PROGRAM HALTED
P	PRINT	PRINT ERROR LOG
R	RESET	PRINT CF'S; INPUT NEW VALUES
T	TERMINATE	TERMINATE THE CURRENT TEST, AND RETURN TO THE PROGRAM EXEC.

NOTES: 1/ AN ILLEGAL COMMAND WILL CAUSE A MESSAGE TO BE PRINTED, LISTING THE VALID COMMANDS THAT MAY BE ENTERED.

10015 NMDU1

FIGURE 1: FAULT TYPES VS. MEMORY TEST PATTERNS

TEST NAME	MAJOR FAULT TYPES TESTED
MODIFIED D=A	1,2,8
STUCK ADR. BIT	1,2
MARCHING 1/0	2,8
GALLOPING COL.	4,5
GALLOPING ROWS	3,5,7
GALLOPING DIAG.	3,4,6,7
GALLOP'G PATTERN	1,2,3,4,6,7,8

KEY:

1: FAULTY ADDRESS DECODING- WRONG CELL OR CHIP ACCESSED
2: FAULTY ADDRESS DECODING- CELL INACCESSIBLE
3: INTERACTIVE COUPLING WITHIN ROWS
4: INTERACTIVE COUPLING WITHIN COLUMNS
5: REFRESH SENSITIVITY
6: SLOW ACCESS TIME
7: SENSE AMPLIFIER RECOVERY
8: CELL FUNCTIONALITY

10016 NMDU1

FIGURE 2: ERROR LOG OUTPUT

CHIP LOCATION				TEST	FAILURE
BOARD	BANK	MOD	BIT	NAMES	COUNT
1	1	C	9	000140	360
1	1	C	12	000140	360
1	1	D	0	000140	32157
1	0	A	6	000202	3
2	0	B	9	000040	64
2	0	B	10	000040	64
2	0	B	15	000040	64

NOTES:

1/ 'TEST NAME(S)' IS A BIT PATTERN. A BIT SET MEANS THAT AN ERROR WAS OCCURRED UNDER THE CORRESPONDING TEST LISTED IN TABLE 1 FOR THE SPECIFIED CHIP LOCATION. (E.G., 41= BITS 10,15 => GALLOPING ROWS AND MODIFIED DATA=ADDRESS, IF RUNNING THE MAPPED VERSION). BIT 0 SET MEANS THAT ERCC WAS ENABLED DURING TESTING.

2/ THE TEST NAMES LISTED INDICATE UNDER WHAT TESTS AN ERROR WAS OCCURRED IN THE WORD.

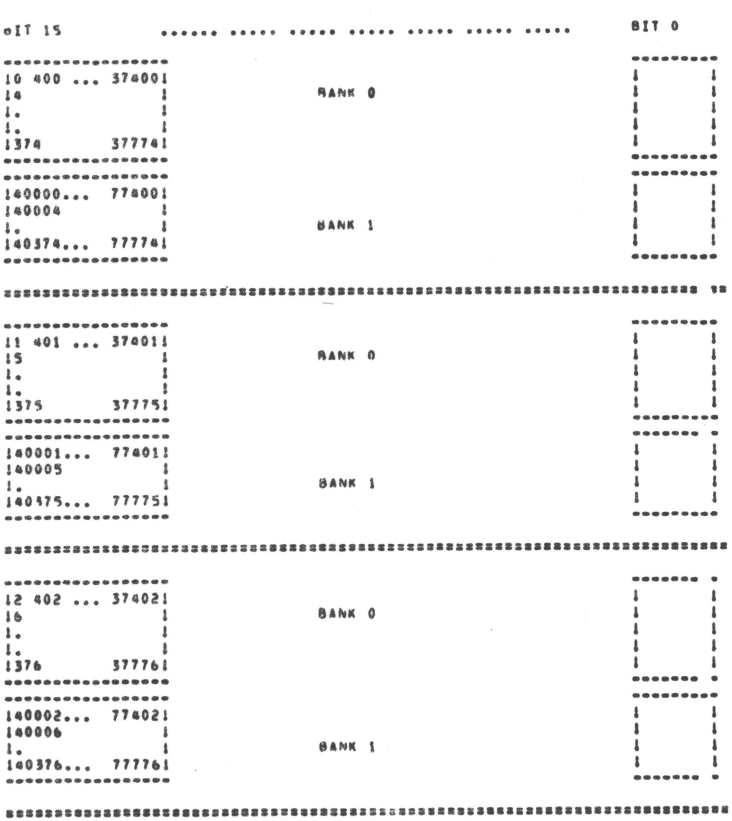
3/ 'FAILURE COUNT' IS THE NUMBER OF TIMES AN ERROR WAS OCCURRED AT THE SPECIFIED CHIP LOCATION.
IF THE FAILURE COUNT=177777, THEN THERE WAS AN OVERFLOW IN THE FAILURE COUNT DATA WORD (I.E., THE NUMBER OF ERRORS FOR THAT CHIP WAS >=177777).

4/ ALL NUMBERS ARE IN OCTAL.

5/ TO GET A HARD COPY OF THE ERROR LOG PRINTOUT, IT IS NECESSARY TO SET SWREG'S BIT 5 BEFORE ENTERING THE PROGRAM MONITOR (UNLESS, OF COURSE, THE CONSOLE DEVICE IS A PRINTER OR TELETYPE). IF THE LOG IS TO BE PRINTED OUT ONLY AT THE CONCLUSION OF THE PROGRAM, SET BIT 5 ANYTIME DURING THE EXECUTION OF THE PROGRAM.

6/ A 'GALLOPING PATTERN' FAILURE COUNT OF 1 SHOULD BE INTERPRETED AS A HARD ERROR. ("1" IS THE MAXIMUM FAILURE COUNT THAT GALPAT WILL GENERATE).

FIGURE 3: MEMORY TOPOLOGY, 4K RAMS



EACH BLOCK REPRESENTS A CHIP. THE NUMBERS WITHIN THE BLOCK DENOTE PHYSICAL ADDRESSES.

THIS IS THE VIEW IF THE FRONT END (I.E., THE ONE WITH THE PINS) IS TOWARDS YOU. THE BANKS ALTERNATE; THE MODULES RUN BACK TO FRONT (A-D); THE BIT NUMBERS RUN RIGHT TO LEFT. IF THERE ARE ANY ERCC CHIPS, THEY WOULD BE APPENDED TO THE LEFT.

A 16K BOARD WOULD HAVE ONLY BANK 0, A 32K BOARD WOULD HAVE BOTH BANK 0 AND BANK 1.

.ENDC
.EJEC

```
;*****************************************************************
;
; NAME: DGCFP.TX                          PART NUMBER: 097-001130
;
;
; DESCRIPTION: DGC NOVA FLOATING POINT TEST PROGRAM
;
;
; REVISION HISTORY:
;
;       REV.            DATE
;
;       00               12/27/78
;
;
; COPYRIGHT (C) DATA GENERAL CORPORATION, 1978
; ALL RIGHTS RESERVED.
; LICENSED MATERIAL - PROPERTY OF DATA GENERAL CORPORATION.
;
;*****************************************************************
```

N4FLTPT


```
;1.0    PROGRAM NAMES
;
;       DGCFP.TX        097-001130
;       DGCFP.SR        094-001533
;       DGCFP.SV        095-001130
;       DGCFP.LS        096-001130
;
;2.0    REVISION HISTORY
;
;       DATE      REVISION
;       ----      --------
;       12/27/78     00
;
;3.0    MACHINE REQUIREMENTS
;
;       1. DGC NOVA IV CENTRAL PROCESSOR
;       2. FIXED POINT MULTIPLY/DIVIDE OPTION
;       3. 8K WORDS READ/WRITE MEMORY
;       4. DGC NOVA FLOATING POINT PROCESSOR
;       5. BASIC I/O TTY INTERFACE AND CONTROL
;
;4.0    SUMMARY
;
;       THIS PROGRAM IS A FUNCTIONAL TEST OF ALL OPERATIONS
;       PERFORMED BY THE DGC NOVA FLOATING POINT ARITHMETIC
;       UNIT.  IT IS MADE UP OF MANY SUBTESTS WHICH INCREASE
;       IN COMPLEXITY FROM SIMPLE STATUS AND INTERRUPT TESTS
;       THROUGH EXERCISERS OF ARITHMETIC OPERATIONS USING
;       RANDOM NUMBERS AND SOFTWARE SIMULATORS.
;
;5.0    OPERATING PROCEDURE
;
;       THIS PROGRAM MAY BE RUN STAND-ALONE OR UNDER A DTOS
;       OR DDOS OPERATING SYSTEM.  THE TEST STARTS AT ADDRESS
;       200 (OCTAL), PRINTS OUT THE PROGRAM TITLE, AND AT
;       THE CONCLUSION OF EACH PASS PRINTS THE NUMBER OF
;       PASSES COMPLETED.  THE STANDARD DTOS SWITCHES ARE
;       USED TO ENABLE LOOPING ON ERROR, PRINTING THE ERROR
;       SUMMARY REPORT, ETC. AS DESCRIBED IN SECTION 8.0.
;       CAT/KITTEN MAY BE USED IF A DISK OR I/O TESTER
;       IS AVAILABLE.  ERROR PRINTOUTS ARE DESCIBED IN
;       SECTION 6.0 AND THE ORDER OF TESTING IS DESCRIBED
;       IN SECTION 7.0.
;
;6.0    ERROR DESCRIPTION
;
;       EACH POSSIBLE ERROR IN THIS TEST HAS BEEN GIVEN A
;       UNIQUE ERROR NUMBER, AND A RECORD OF ALL FAILURES
;       ENCOUNTERED IS MAINTAINED BY THE ERROR HANDLER
;       SUBROUTINE.  THIS ALLOWS OVERNIGHT RUNS OF THE TEST
;       WITH LOOPING ON ERROR AND PRINT DISABLED WHILE
;       STILL INDICATING WHICH SUBTESTS, IF ANY, FAILED
;       THROUGH A SUMMARY REPORT THE NEXT MORNING.  THIS
;       RECORD IS CLEARED WHEN THE TEST IS RESTARTED OR
;       WHEN SWITCH 7 IS TURNED ON AND A SUMMARY REPORT
;       REQUESTED.  EITHER THE FIRST TIME THAT AN ERROR
;       IS ENCOUNTERED OR WHENEVER SWITCH 8 IS ON, IF THE
;       CONSOLE AND/OR LINE PRINTER ARE ENABLED A DETAILED
;       ERROR REPORT WILL BE PRINTED IN THE FOLLOWING
;       FORMAT:
```

```
01              ;
02              ;       ERROR NUMBER XXX         ENCOUNTERED SUBTEST XXX
03              ;       CRY     AC0     AC1     AC2     AC3     PC
04              ;       X       XXXXXX  XXXXXX  XXXXXX  XXXXXX  XXXXXX
05              ;
06              ;       REFERENCE IN THE LISTING TO THE SUBTEST AND ERROR
07              ;       NUMBER INDICATED WILL EXPLAIN EXACTLY WHAT WAS
08              ;       BEING TESTED AND THE MEANING OF THE ACCUMULATOR
09              ;       CONTENTS.  IN CASES WHERE FLOATING POINT ARITHMETIC
10              ;       CALCULATIONS ARE BEING MADE, AN ADDITIONAL PRINTOUT
11              ;       MAY BE PROVIDED, GIVING THE FULL FLOATING POINT
12              ;       NUMBERS CONCERNED IN HEX FORMAT.  IN THOSE CASES
13              ;       WHERE RANDOM NUMBERS ARE USED AS OPERANDS, THESE
14              ;       OPERANDS WILL ALSO BE PRINTED.  AN EXAMPLE OF ONE
15              ;       OF THESE PRINTOUTS FOLLOWS:
16              ;
17              ;       FLOATING POINT DATA IN HEX FORMAT FOLLOWS:
18              ;                       S EX  MANT              S EX  MANT
19              ;       OPERAND A       X XX XXXXXX OPERAND B   X XX XXXXXX
20              ;       EXP. RESULT X XX XXXXXX ACT. RESULT X XX XXXXXX
21              ;
22              ;7.0    TEST DESCRIPTION
23              ;
24              ;       THIS PROGRAM IS DESIGNED TO PROCEED FROM VERY SIMPLE
25              ;       TESTS OF STATUS AND REGISTERS THROUGH MORE AND MORE
26              ;       COMPLEX OPERATIONS UNTIL ALL FUNCTIONS HAVE BEEN
27              ;       CHECKED.  THE OPERATIONS CHECKED BY VARIOUS TESTS
28              ;       ARE DETAILED BELOW.  ALL TEST NUMBERS ARE IN OCTAL
29              ;       AS THEY APPEAR IN THE LISTING.
30              ;
31              ;       TESTS           FUNCTIONS CHECKED
32              ;       -----           --------- -------
33              ;
34              ;       1-10            READ & WRITE STATUS, INTERRUPTS
35              ;       11-21           CLEAR, LOAD EXP., READ HIGH WORD
36              ;       22-23           STORE FPAC
37              ;       24-26           LOAD FPAC
38              ;       27-30           COPY FPAC => TEMP, TEMP => FPAC
39              ;       31              NEGATE, ABSOLUTE VALUE
40              ;       32-50           NORMALIZE
41              ;       51-73           SCALE, CHECK GUARD DIGIT
42              ;       74-104          ADD SINGLE AND DOUBLE
43              ;       105-111         SUBTRACT SINGLE AND DOUBLE
44              ;       112-121         MULTIPLY SINGLE AND DOUBLE
45              ;       122-131         DIVIDE SINGLE AND DOUBLE
46              ;
47
48
49              ;8.     SWITCH SETTINGS
50              ;
51              ;       LOCATION "SWREG" IS USED TO SELECT THE PROGRAM OPTIONS
52              ;       (NOT SYSTEM CONFIGURATION). WHILE RUNNING UNDER DTOS,
53              ;       THIS LOCATION WILL BE LOADED BY THE MONITOR.
54              ;       HOWEVER UNDER STAND ALONE AND PROGRAM LOAD MODES THIS
55              ;       LOCATION WILL BE SET ACCORDING TO THE ANSWERS SUPPLIED
56              ;       BY THE OPERATOR. IN ANY CASE THE OPTIONS CAN BE CHANGED
57              ;       OR VERIFIED BY USING ONE OF THE COMMANDS GIVEN IN SEC.
58              ;       8.2
59              ;
60              ;
```



```
01              ;8.1    SWITCH OPTIONS
02              ;       DIFFERENT BITS AND THEIR INTERPRETATION AT LOCATION
03              ;       "SWREG" IS AS FOLLOWS:
04              ;
05              ;       BIT     OCTAL   BINARY  INTERPRETATION
06              ;               VALUE   VALUE
07              ;
08              ;       1               0       LOOP ON ERROR
09              ;               40000   1       SKIP LOOPING ON ERROR
10              ;
11              ;       2               0       PRINT TO CONSOLE
12              ;               20000   1       ABORT PRINT OUT TO CONSOLE
13              ;
14              ;       3               0       DO NOT PRINT % FAILURE
15              ;               10000   1       PRINT % FAILURE
16              ;
17              ;       4               0       ALLOW END OF PASS PRINT OUT
18              ;               04000   1       SUPPRESS END OF PASS PRINT OUT
19              ;
20              ;       5               0       DO NOT PRINT ON THE LINE PRINTER
21              ;               02000   1       PRINT ON THE LINE PRINTER
22              ;
23              ;       6               0       DO NOT HALT ON ERROR
24              ;               01000   1       HALT ON ERROR
25              ;
26              ;       7               0       DO NOT PRINT SUMMARY AND/OR
27              ;                               PASSING OF EACH SUBTEST
28              ;               00400   1       PRINT SUMMARY AND/OR
29              ;                               PASSING OF EACH SUBTEST
30              ;       8               0       PRINT ONLY THE FIRST ERROR
31              ;               00200   1       PRINT EVERY ERROR
32              ;
33              ;8.2    SWITCH COMMANDS
34              ;       ONCE THE PROGRAM STARTS EXECUTING THE STATE OF ANY OF
35              ;       THE BITS CAN BE CHANGED BY HITTING KEYS 1-9, A-F. THE
36              ;       PROGRAM WILL CONTINUE RUNNING AFTER UPDATING THE OPTIONS.
37              ;       EACH KEY WILL COMPLEMENT THE STATE OF THE BIT AFFILIAT-
38              ;       ED WITH IT, THUS BIT 4 CAN BE ALTERED BY HITTING KEY 4.
39              ;       SETTING OF ANY BIT OF LOCATION "SWREG" WILL SET BIT 0.
40              ;       (DEFAULT MODE IS DEFINED AS ALL BITS OF SWREG SET TO 0)
41              ;       THE PROGRAM CAN BE LOCKED INTO SWITCH MODIFICATION MODE
42              ;       BY TYPING A 0, IN WHICH CASE MORE THAN ONE BIT CAN BE
43              ;       CHANGED BEFORE  CONTROL IS ALLOWED TO RETURN TO THE
44              ;       MAIN PROGRAM.
45              ;
46              ;8.2.1  OTHER COMMANDS
47              ;
48              ;       "CR"    A "RETURN" CAN BE TYPED TO CONTINUE THE PROGRAM
49              ;               AFTER ITS LOCKED IN A SWITCH MODIFICATION MODE
50              ;
51              ;       ^D      THIS COMMAND GIVEN AT ANY TIME WILL RESET "SWREG"
52              ;               TO DEFAULT MODE AND RESTART THE PROGRAM.
53              ;
54              ;       ^R      THIS COMMAND GIVEN AT ANY TIME WILL RESTART THE
55              ;               PROGRAM. SWITCHES ARE LEFT WITH THE VALUES THEY
56              ;               HAD BEFORE THE COMMAND WAS ISSUED.
57              ;
58              ;       ^O      THIS COMMAND GIVEN AT ANY TIME WILL CAUSE THE
59              ;               PROGRAM CONTROL TO GO TO ODT (NOTE: THIS IS AN
60              ;               OPTIONAL COMMAND AND IS AVAILBLE ONLY IF
```

```
01              ;               ODTPK IS PRESENT)
02              ;
03              ;       M       THIS COMMAND GIVEN AT ANY TIME WILL PRINT THE
04              ;               CURRENT OPERATING MODES.
05              ;
06
07              ;
08
09              ;9.     OCTAL DEMUG TOOL (ODT)
10              ;
11              ;       THE DIAGNOSTIC IS EQUIPED WITH A BUILT IN ODT WHICH CAN
12              ;       BE ACCESSED BY HITTING CONTROL @ (^@) AT ANY TIME DURING
13              ;       THE EXECUTION OF THE PROGRAM (AFTER SETTING THE PARA-
14              ;       METERS).
15              ;       ON ENTERING ODT THE ADDRESS OF THE LOCATION HAVING THE
16              ;       NEXT INSTRUCTION TO BE EXECUTED WILL BE TYPED-OUT.
17              ;
18              ;9.1    CONVENTIONS AND SYMBOLS
19              ;       THE FOLLOWING CONVENTIONS ARE USED BY THE ODT:
20              ;       ?       PRESSING ANY ILLEGAL KEY CAUSES THE ODT TO RES-
21              ;               POND WITH A "?".
22              ;       @       ODT IS READY AND AT YOUR SERVICE.
23              ;
24              ;9.2    COMMAND STRUCTURE
25              ;       AN ODT COMMAND HAS THE FOLLOWING FORMAT:
26              ;               (ARGUMENT)(COMMAND)
27              ;       AN ARGUMENT MAY BE ONE OF THE FOLLOWING:
28              ;       "EXP"   AN OCTAL EXPRESSION CONSISTING OF OCTAL NUMBERS
29              ;               SEPARATED BY PLUS (+) OR MINUS (-) SIGNS. LEAD-
30              ;               ING ZEROS NEED NOT BE TYPED.
31              ;       "ADR"   AN ADDRESS IS THE SAME AS AN EXPRESSION EXCEPT
32              ;               THAT BIT 0 IS NEGLECTED.
33              ;       A COMMAND IS A SINGLE TELETYPE CHARACTER
34              ;
35              ;9.3    ODT COMMANDS
36              ;       THE LOCATIONS THAT CAN BE EXAMINED AND MODIFIED BY THE
37              ;       USER ARE CALLED CELLS. THESE CELLS ARE OF TWO TYPES:
38              ;       INTERNAL CPU CELLS AND MEMORY LOCATIONS.
39              ;
40              ;9.3.1  OPENING INTERNAL CELLS
41              ;       THE COMMAND TO OPEN ONE OF THE INTERNAL REGISTERS IS OF
42              ;       THE FORM "NA" WHERE N IS ANY OCTAL EXPRESSION BETWEEN
43              ;       0 AND 7
44              ;       0-3     FOR ACCUMULATORS 0-3
45              ;       4       FOR PC OF THE NEXT INSTRUCTION TO BE EXECUTED IN
46              ;               THE EVENT OF A "P" COMMAND.
47              ;       5       CPU AND TTO STATUS
48              ;               BIT     INTERPRETATION
49              ;               15      STATUS OF TTO DONE FLAG
50              ;               14      STATUS OF INTERRUPTS (ION FLAG)
51              ;               13      STATUS OF CARRY BIT
52              ;       6       ADDRESS OF THE LOCATION HAVING THE BREAK POINT (IF
53              ;               ANY)
54              ;       7       INSTRUCTION AT THE BREAK POINT LOCATION
55
56              ;       OTHER COMMANDS TO OPEN CELLS ARE:
57              ;
58              ;       "ADR"/  OPEN THE CELL AND PRINT ITS CONTENTS
59              ;       ./      OPEN THE CELL CURRENTLY POINTED TO BY THE POINTER
60              ;               AND PRINT ITS CONTENTS.
```



```
01              ;       .+"ADR"/        ADD "ADR" TO THE POINTER, OPEN THE CELL
02              ;               AND PRINT ITS CONTENTS.
03              ;       .-"ADR"/        SUBTRACT "ADR" FROM THE POINTER, OPEN
04              ;               THE CELL AND PRINT ITS CONTENTS.
05              ;       "CR"    THE RETURN KEY IS USED TO CLOSE THE OPEN CELL
06              ;               WITH OR WITHOUT MODIFICATION.
07              ;       "LF"    LINE FEED IS USED TO CLOSE THE OPEN CELL WITH OR
08              ;               WITHOUT MODIFICATION AND TO OPEN THE SUCCEEDING
09              ;               CELL.
10              ;       ^       CLOSE THE OPEN CELL WITH OR WITHOUT MODIFICATION
11              ;               AND OPEN THE PRECEDING CELL
12              ;       /       CLOSE THE OPEN CELL WITHOUT MODIFICATION, AND
13              ;               OPEN THE CELL POINTED TO BY ITS CONTENTS.
14              ;       +"ADR"/ CLOSE THE OPEN CELL WITHOUT MODIFICATION, AND
15              ;               OPEN THE CELL POINTED TO BY ITS CONTENTS + "ADDR".
16              ;       -"ADR"/ CLOSE THE OPEN CELL WITHOUT MODIFICATION, AND
17              ;               OPEN THE CELL POINTED TO BY ITS CONTENTS - "ADR".
18              ;
19              ;9.3.2  MODIFICATION OF A CELL
20              ;       ONCE A CELL HAS BEEN OPENED ITS CONTENTS CAN BE MODIFIED
21              ;       BY TYPING THE NEW VALUE THE CELL IS TO CONTAIN IN THE
22              ;       FORM OF AN OCTAL EXPRESSION FOLLOWED BY "CR" OR "LF".
23              ;       IF A + OR - IS TYPED AS THE FIRST CHARACTER OF THE EX-
24              ;       PRESSION THEN THE VALUE OF THE EXPRESSION IS ADDED TO OR
25              ;       SUBTRACTED FROM THE OLD CONTENTS OF THE CELL. THE
26              ;       ADDRESS ITSELF OR AN EXPRESION RELATIVE TO THE ADDRESS
27              ;       CAN BE DEPOSITED BY TYPING A "." OR ".+/-OCTAL EXPRESS-
28              ;       ION". A RUBOUT COMMAND GIVEN RIGHT AFTER OPENING A CELL
29              ;       ALLOWS THE MODIFICATION OF ITS CONTENTS AS IF THEY WERE
30              ;       TYPED IN JUST BEFORE THE COMMAND WAS ISSUED.
31              ;
32              ;9.3.3  OTHER ODT COMMANDS
33              ;       RUBOUT  THIS KEY IS USED TO DELETE ERRONEOUSLY TYPED
34              ;               DIGITS. EACH TIME THE KEY IS PRESSED THE RIGHT MOST
35              ;               DIGIT IS DELETED AND ECHOED ON THE TERMINAL. IF
36              ;               THE RUBOUT KEY IS PRESSED RIGHT AFTER OPENING A
37              ;               CELL THEN IT DELETES THE RIGHT MOST DIGIT OF THE CELLS
38              ;               CONTENTS. THIS ALLOWS THE MODIFICATION OF THE CELL
39              ;               AS IF ITS CONTENTS WERE TYPED IN JUST BEFORE THE
40              ;               KEY WAS PRESSED.
41              ;       "ADR"B  INSERT A BREAK POINT AT LOCATION "ADR".
42              ;               ONLY ONE BREAK POINT CAN BE INSERTED AND ANY
43              ;               ENTRY TO ODT AFTER EXECUTING A BREAK POINT WILL
44              ;               CAUSE IT TO BE DELETED.
45              ;       0       DELETE THE BREAK POINT IF ANY.
46              ;       P       RESTART THE EXECUTION OF THE PROGRAM AT LOCATION
47              ;               POINTED BY 4A.
48              ;       "ADR"R  START EXECUTING THE PROGRAM AT "ADR" AFTER AN
49              ;               IO-RESET.
50              ;       K       KILL THE STRING TYPED SO FAR. THE ODT RESPONDS
51              ;               WITH A "?" AND THE OPEN CELL IS CLOSED WITHOUT
52              ;               MODIFICATION.
53              ;       =       PRINT THE OCTAL VALUE OF THE INPUT ONLY.
54              ;               THIS WILL CLOSE ANY OPEN CELLS WITHOUT
55              ;               MODIFICATION AND WILL NOT OPEN A CELL
56              ;
57              ;
58              ;       NOTE:   IN PROGRAMS WHICH RELOCATE THEMSELVES THE
59              ;               THE USER SHOULD PLACE BREAK POINTS ONLY IN THE
60              ;               THE ORIGINAL PROGRAM AREA. IF A BREAK POINT IS
```

```
0001 N4MPD    MACRO REV 06.30          00:12:43 12/22/78
01
02
03
04
05
06                                        1
07              ;*****************************************************************
08              ;
09              ; NAME: N4MPDIAG.TX                        PART NUMBER: 097-001131
10              ;
11              ;
12              ; DESCRIPTION: NOVA 4 MMPU DIAGNOSTIC
13              ;
14              ;
15              ; REVISION HISTORY:
16              ;
17              ;       REV.            DATE
18              ;
19              ;       00              12/29/78
20              ;
21              ;
22              ; COPYRIGHT (C) DATA GENERAL CORPORATION, 1978
23              ; ALL RIGHTS RESERVED.
24              ;*****************************************************************
25
26
27
```

N4 MMPU

```
10002 N4MPD
01
02              ;N4MPDIAG-NOVA4 MMPU DIAGNOSTIC
03              ;
04              ;THIS DIAGNOSTIC IS DESIGNED TO RUN IN AN
05              ;AUTO-LOAD AUTO-RUN ENVIRONMENT.
06              ;
07              ;;0.    REVISION HISTORY
08              ;;              REV.00 WAS CREATED FROM REV 03 OF N3MMPU.
09              ;;              SLIGHT MODIFICATIONS WERE NECESSARY TO RUN
10              ;;              ON THE NOVA 4.
11              ;;
12              ;1.0    ABSTRACT
13              ;               THIS TEST IS
14              ;               DESIGNED TO VERIFY THE OPERATION OF THE
15              ;               MEMORY MANAGEMENT UNIT (MMU) FEATURE.
16              ;               THIS PROGRAM IS A PREREQUSITE TO THE
17              ;               MULTIPROGRAMMING RELIABILITY TEST PROGRAM.
18              ;
19              ;2.0    MACHINE REQUIREMENTS
20              ;       2.1     NOVA 4 PROCESSOR WITH MMU OPTION.
21              ;       2.2     4K OF READ/WRITE MEMORY.
22              ;       2.3     TTY OR CRT
23              ;       2.4     RTC (OPTIONAL)
24              ;       2.5     I/O TESTER (OPTIONAL)
25              ;       2.6     FLOATING POINT UNIT(OPTIONAL)
26              ;
27              ;3.0    OPERATING PROCEDURE
28              ;       3.1     LOADING
29              ;               LOAD PROGRAM VIA THE BINARY LOADER.
30              ;       3.2     START ADDRESS
31              ;               TYPE 200R ON CONSOLE.
32              ;               PRESS NEW-LINE.
33              ;               THE PROGRAM STARTS BY PRINTING OUT THE
34              ;               PROGRAM NAME AND REVISION NUMBER. IT
35              ;               THEN INFORMS THE OPERATOR OF THE EXISTENCE
36              ;               OF THE RTC,I/O TESTER,AND FPU. THIS IS FOLLOWED
37              ;               BY THE MEMORY SIZE INFORMATION BOTH WITHOUT AND
38              ;               WITH THE MAP FEATURE BEING USED.
39              ;
```

S?MPD 3.3

;3.3. 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.
; 3.3.2

;3.3.1 SWITCH OPTIONS
; DIFFERENT BITS AND THEIR INTERPRETATION AT LOCATION
; "SWREG" IS AS FOLLOWS:

BIT	OCTAL VALUE	BINARY VALUE	INERPRETATION
1		0	LOOP ON ERROR
	40000	1	SKIP LOOPING ON ERROR
2		0	PRINT TO CONSOLE
	20000	1	ABORT PRINT OUT TO CONSOLE
3		0	DO NOT PRINT % FAILURE
	10000	1	PRINT % FAILURE
4		0	ALLOW END OF PASS PRINT OUT
	04000	1	SUPPRESS END OF PASS PRINT OUT
5		0	DO NOT PRINT ON THE LINE PRINTER
	02000	1	PRINT ON THE LINE PRINTER
6		0	DO NOT HALT ON ERROR
	01000	1	HALT ON ERROR
7		0	DO NOT PRINT SUMMARY AND/OR PASSING OF EACH SUBTEST
	00400	1	PRINT SUMMARY AND/OR PASSING OF EACH SUBTEST
8		0	PRINT ONLY THE FIRST ERROR
	00200	1	PRINT EVERY ERROR

;3.3.2 SWITCH COMMANDS
; 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 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.
;

;3.3.2.1 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 AVAILBLE ONLY IF
; ODTPK IS PRESENT)
;
; M THIS COMMAND GIVEN AT ANY TIME WILL PRINT THE
; CURRENT OPERATING MODES.
;
;
;
; 3.3 KEY ENTERED OPTIONS
;
; KEY 1 (0) = LOOP ON ERROR
; KEY 1 (1) = PROCEED FROM ERROR
; KEY 2 (1) = INHIBIT PRINTOUT
; KEY 3 (1) = PRINT FAILURE RATE
; KEY 4 (1) = DELETE PRINT OF "PASS XX"MESSAGE
; KEY 5 (1) = OPTIONAL LINE PRINTER OUTPUT
; KEY 6 (1) = HALT ON ERROR
;
; KEY M = PRINT CONTENTS OF SWREG
;
; KEY (C)D = SET SWREG TO DEFAULT MODE AND
; RESTART PROGRAM.
;
; KEY (C)R = RESTART PROGRAM WITH CURRENT OPTIONS.
; IF THE SYSTEM DOESN'T CONTAIN A TERMINAL/TTY THEN
; OPTIONS MAY BE SELECTED BY PATCHING INTO LOCATION
; "SWREG".
;
; 3.4 NORMAL OPERATION
; PROGRAM WILL EXECUTE ALL TESTS IN SEQUENCE
; AND AUTOMATICALLY LOOP.
; IF A DEVICE IS FOUND TO NOT EXIST THE
; TEST USING THIS DEVICE WILL BE BYPASSED.
; A MESSAGE "PASS " WILL BE PRINTED AT THE
; END OF EACH SUCESSFUL PASS IF OPTION 4
; IS NOT SELECTED.
;
;

O?DTD 3.5

;3.5. OCTAL DEBUG TOOL (ODT)
;
; THE DIAGNOSTIC IS EQUIPED WITH A BUILT IN ODT WHICH CAN
; BE ACCESSED BY HITTING CONTROL O (^O) AT ANY TIME DURING
; THE EXECUTION OF THE PROGRAM (AFTER SETTING THE PARA-

```
01              ;       METERS).
02              ;       ON ENTERING ODT THE ADDRESS OF THE LOCATION HAVING THE
03              ;       NEXT INSTRUCTION TO BE EXECUTED WILL BE TYPED-OUT.
04              ;
05              ;3.5.1  CONVENTIONS AND SYMBOLS
06              ;       THE FOLLOWING CONVENTIONS ARE USED BY THE ODT:
07              ;       ?       PRESSING ANY ILLEGAL KEY CAUSES THE ODT TO RES-
08              ;               POND WITH A "?".
09              ;       @       ODT IS READY AND AT YOUR SERVICE.
10              ;
11              ;3.5.2  COMMAND STRUCTURE
12              ;       AN ODT COMMAND HAS THE FOLLOWING FORMAT:
13              ;               [ARGUMENT][COMMAND]
14              ;       AN ARGUMENT MAY BE ONE OF THE FOLLOWING:
15              ;       "EXP"   AN OCTAL EXPRESSION CONSISTING OF OCTAL NUMBERS
16              ;               SEPARATED BY PLUS (+) OR MINUS (-) SIGNS. LEAD-
17              ;               ING ZEROS NEED NOT BE TYPED.
18              ;       "ADR"   AN ADDRESS IS THE SAME AS AN EXPRESSION EXCEPT
19              ;               THAT BIT 0 IS NEGLECTED.
20              ;       A COMMAND IS A SINGLE TELETYPE CHARACTER
21              ;
22              ;3.5.3  ODT COMMANDS
23              ;       THE LOCATIONS THAT CAN BE EXAMINED AND MODIFIED BY THE
24              ;       USER ARE CALLED CELLS. THESE CELLS ARE OF TWO TYPES:
25              ;       INTERNAL CPU CELLS AND MEMORY LOCATIONS.
26              ;
27              ;3.5.3.1        OPENING INTERNAL CELLS
28              ;       THE COMMAND TO OPEN ONE OF THE INTERNAL REGISTERS IS OF
29              ;       THE FORM "NA" WHERE N IS ANY OCTAL EXPRESSION BETWEEN
30              ;       0 AND 7
31              ;       0-3     FOR ACCUMULATORS 0-3
32              ;       4       FOR PC OF THE NEXT INSTRUCTION TO BE EXECUTED IN
33              ;               THE EVENT OF A "P" COMMAND.
34              ;       5       CPU AND TTO STATUS
35              ;               BIT     INTERPRETATION
36              ;               15      STATUS OF TTO DONE FLAG
37              ;               14      STATUS OF INTERRUPTS (ION FLAG)
38              ;               13      STATUS OF CARRY BIT
39              ;       6       ADDRESS OF THE LOCATION HAVING THE BREAK POINT (IF
40              ;               ANY)
41              ;       7       INSTRUCTION AT THE BREAK POINT LOCATION
42              ;
43              ;       OTHER COMMANDS TO OPEN CELLS ARE:
44              ;
45              ;       "ADR"/  OPEN THE CELL AND PRINT ITS CONTENTS
46              ;       ./      OPEN THE CELL CURRENTLY POINTED TO BY THE POINTER
47              ;               AND PRINT ITS CONTENTS.
48              ;       .+"ADR"/        ADD "ADR" TO THE POINTER, OPEN THE CELL
49              ;               AND PRINT ITS CONTENTS.
50              ;       .-"ADR"/        SUBTRACT "ADR" FROM THE POINTER, OPEN
51              ;               THE CELL AND PRINT ITS CONTENTS.
52              ;       "CR"    THE RETURN KEY IS USED TO CLOSE THE OPEN CELL
53              ;               WITH OR WITHOUT MODIFICATION.
54              ;       "LF"    LINE FEED IS USED TO CLOSE THE OPEN CELL WITH OR
55              ;               WITHOUT MODIFICATION AND TO OPEN THE SUCCEEDING
56              ;               CELL.
57              ;       ^       CLOSE THE OPEN CELL WITH OR WITHOUT MODIFICATION
58              ;               AND OPEN THE PRECEDING CELL
59              ;       /       CLOSE THE OPEN CELL WITHOUT MODIFICATION, AND
60              ;               OPEN THE CELL POINTED TO BY ITS CONTENTS.
```



```
01              ;       +"ADR"/ CLOSE THE OPEN CELL WITHOUT MODIFICATION, AND
02              ;               OPEN THE CELL POINTED TO BY ITS CONTENTS + "ADDR".
03              ;       -"ADR"/ CLOSE THE OPEN CELL WITHOUT MODIFICATION, AND
04              ;               OPEN THE CELL POINTED TO BY ITS CONTENTS - "ADR".
05              ;
06              ;3.5.3.2        MODIFICATION OF A CELL
07              ;       ONCE A CELL HAS BEEN OPENED ITS CONTENTS CAN BE MODIFIED
08              ;       BY TYPING THE NEW VALUE THE CELL IS TO CONTAIN IN THE
09              ;       FORM OF AN OCTAL EXPRESSION FOLLOWED BY "CR" OR "LF".
10              ;       IF A + OR - IS TYPED AS THE FIRST CHARACTER OF THE EX-
11              ;       PRESSION THEN THE VALUE OF THE EXPRESSION IS ADDED TO OR
12              ;       SUBTRACTED FROM THE OLD CONTENTS OF THE CELL. THE
13              ;       ADDRESS ITSELF OR AN EXPRESION RELATIVE TO THE ADDRESS
14              ;       CAN BE DEPOSITED BY TYPING A "." OR ".+/-OCTAL EXPRESS-
15              ;       ION". A RUBOUT COMMAND GIVEN RIGHT AFTER OPENING A CELL
16              ;       ALLOWS THE MODIFICATION OF ITS CONTENTS AS IF THEY WERE
17              ;       TYPED IN JUST BEFORE THE COMMAND WAS ISSUED.
18              ;
19              ;3.5.3.3        OTHER ODT COMMANDS
20              ;       RUBOUT  THIS KEY IS USED TO DELETE ERRONEOUSLY TYPED
21              ;               DIGITS. EACH TIME THE KEY IS PRESSED THE RIGHT MOST
22              ;               DIGIT IS DELETED AND ECHOED ON THE TERMINAL. IF
23              ;               THE RUBOUT KEY IS PRESSED RIGHT AFTER OPENING A
24              ;               CELL THEN IT DELETES THE RIGHT MOST DIGIT OF THE CELLS
25              ;               CONTENTS. THIS ALLOWS THE MODIFICATION OF THE CELL
26              ;               AS IF ITS CONTENTS WERE TYPED IN JUST BEFORE THE
27              ;               KEY WAS PRESSED.
28              ;       "ADR"B  INSERT A BREAK POINT AT LOCATION "ADR".
29              ;               ONLY ONE BREAK POINT CAN BE INSERTED AND ANY
30              ;               ENTRY TO ODT AFTER EXECUTING A BREAK POINT WILL
31              ;               CAUSE IT TO BE DELETED.
32              ;       D       DELETE THE BREAK POINT IF ANY.
33              ;       P       RESTART THE EXECUTION OF THE PROGRAM AT LOCATION
34              ;               POINTED BY 6A.
35              ;       "ADR"R  START EXECUTING THE PROGRAM AT "ADR" AFTER AN
36              ;               IO-RESET.
37              ;       K       KILL THE STRING TYPED SO FAR, THE ODT RESPONDS
38              ;               WITH A "?" AND THE OPEN CELL IS CLOSED WITHOUT
39              ;               MODIFICATION.
40              ;       =       PRINT THE OCTAL VALUE OF THE INPUT ONLY.
41              ;               THIS WILL CLOSE ANY OPEN CELLS WITHOUT
42              ;               MODIFICATION AND WILL NOT OPEN A CELL
43              ;
44              ;
45              ;       NOTE:   IN PROGRAMS WHICH RELOCATE THEMSELVES THE
46              ;               THE USER SHOULD PLACE BREAK POINTS ONLY IN THE
47              ;               THE ORIGINAL PROGRAM AREA. IF A BREAK POINT IS
48              ;               PLACED OUTSIDE THIS AREA THE RESULTS WILL
49              ;               BE UNPREDICTABLE.
50              ;
51              ;
```

```
10007 N4MPD
01
02                ;
03                ;4.0    ERROR DESCRIPTION
04                ;       4.1     NORMAL
05                ;               UPON THE DETECTION OF AN ERROR THE PROGRAM WILL
06                ;               PRINT OUT THE CONTENTS OF ACCUMULATOR,ETC.
07                ;               AND IF SWITCHES ARE APPROPRIATE, IT WILL HALT.
08                ;               CONSULT THE
09                ;               LISTING FOR A DETAILED TEST DESCRIPTION.
10                ;               SET THE SWITCHES AS DESIRED AND PRESS CONTINUE.
11                ;       4.2     ABNORMAL
12                ;               THERE ARE SEVERAL TYPES OF UNEXPECTED FAILURES
13                ;               WHICH WILL CAUSE THE PGM TO ENTER ODT.  THEY ARE AS
14                ;               FOLLOWS:
15                ;                       UNEXPECTED INTERRUPT
16                ;                       STACK OVERFLOW
17                ;               THE CAUSE OF ANY OF THESE FAILURES SHOULD BE
18                ;               CORRECTED BEFORE RESUMING TESTING.
19                ;
20                ;5.0    PROGRAM DESCRIPTION
21                ;       5.1     COMMON SUBROUTINE CALLS
22                ;               THE DIAGNOSTIC IS COMPRISED OF A SERIES OF
23                ;               SHORT TESTS.  BASICALLY, EACH TEST CONSISTS
24                ;               OF A SETUP PROCEDURE, ONE OR MORE EVALUATING
25                ;               CASES WITH ERROR CALLS, AND A LOOP CAPABILITY.
26                ;               EACH PARTICULAR TEST CASE IS DESCRIBED IN THE
27                ;               LISTING.  THE COMMON ROUTINES FOR SETUP (SETUP),
28                ;               ERROR CALLS (EHALT), AND LOOP (LOOP) ARE
29                ;               DESCRIBED HERE,ALONG WITH OTHER
30                ;               COMMONLY CALLED ROUTINES.
31                ;
32                ;               SETUP
33                ;               EACH TEST BEGINS WITH A CALL TO SETUP.  THIS
34                ;               ROUTINE SETS UP THE LOOP ADDRESS,
35                ;               RESETS CERTAIN ERROR SWITCHES AND ITERATION
36                ;               COUNTS.  IT ALSO STORES AN ADDRESS OF AN ENTRANCE
37                ;               TO AN EHALT  WITHIN EACH TEST INTO
38                ;               LOCATIONS 3,AND 41 AND INTERRUPT
39                ;                HANDLER ADDRESS INTO LOC 1 .
40                ;
41                ;               EHALT
42                ;               THIS ROUTINE IS CALLED WHEN AN ERROR IS
43                ;               DETECTED. THE CONTENTS OF THE ACCUMULATORS,
44                ;               THE PASS COUNT, CONTENTS OF LOC 0, ARE PRINTED.
```

```
10008 N4MPD
01                ;
02                ;               LOOP
03                ;               THIS ROUTINE IS CALLED AT THE END OF EACH TEST
04                ;               SEQUENCE.  IT IS USED TO ITERATE THE SEQUENCE
05                ;               100 TIMES IF NO ERROR HAS BEEN DETECTED.
06                ;               IF AN ERROR HAS BEEN DETECTED, IT IS USED TO
07                ;               MAINTAIN THE SCOPE LOOP AND INTERROGATE THE
08                ;               SWITCHES, ETC.
09                ;
10                ;
11                ;               LEQPM
12                ;               MAP ALL OF MEMORY TO
13                ;               LOGICAL = PHYSICAL
14                ;               FOR MAP USERS A&B
15                ;
16                ;               LEPMD
17                ;               MAP DCM LOGICAL=PHYSICAL
18                ;               FOR BOTH MAP A&B
19                ;
20                ;               MTVIO
21                ;               CHECK FOR VIOLATION FLAG SET
22                ;
23                ;               STSCK
24                ;               COMPARE SPECIFIER WORD WITH STATUS
25                ;               REGISTER CONTENTS
26                ;
27                ;               CKVLD
28                ;               COMPARE SPECIFIER WORD WITH VIOLATION
29                ;               DATA REG. CONTENTS
30                ;
31                ;               BUFCK
32                ;               COMPARES SPECIFIER WORD WITH THE
33                ;               CONTENTS OF THE DATA BUFFER AREA
34                ;               AC0=ACTUAL
35                ;               AC1=EXPECTED
36                ;               AC2=ADDRESS OF  WORD
37                ;
38                ;               STBUF
39                ;               STORES THE SPECIFIED WORD THRUOUT
40                ;               THE DATA BUFFER AREA
41                ;
42                ;
43                        P2NED   5.1.1
44                ;5.1.1. PRE MACRO
45                ;       THIS MACRO PROVIDES ALL THE NECESSARY PRE-
46                ;       DIAGNOSTIC PROGRAM COMMUNICATIONS WITH DTOS.
47                ;       IT SIZES MEMORY, MONITORS THE EGGS BLOCK,
48                ;       AND PROVIDES TO THE USER THE FOLLOWING:
49                ;
50                ;       1)      THE LOGICAL TOP OF MEMORY
51                ;
52                ;       2)      THE TOPMOST MEMORY LOCATION AVAILABLE
53                ;               FOR THE DIAGNOSTIC PROGRAM
54                ;
55                ;       3)      THE HIGHEST 1K BLOCK OF MEM (IF MAPPED)
56                ;
57                ;       4)      A STATUS WORD CONTAINING THE STATE OF
58                ;               THE CATSWITCH, THE MODE OF OPERATION,
59                ;               AND THE TEST PROGRAM PASS COUNT
60                ;
```

```
01              ;       5)      A STATUS WORD CONTAINING, CPU TYPE,
02              ;               MAP STATUS, (MAP=NO MAP), AND THE
03              ;               MAP TYPE, IF APPLICABLE.
04              ;
05              ;       6)      A POINTER TO THE I/O MODULE
06              ;
07              ;       7)      A POINTER TO THE CAT/KITTEN PROGRAM
08              ;               IF DESIRED.
09              ;
10              ;5.1.1.1        THE USER MUST ALLOCATE PAGE ZERO STORAGE FOR
11              ;       THESE PARAMETERS, AND DEFINE THEM IN THE MACRO
12              ;       CALL.
13              ;
14              ;
15              ;5.1.1.2        CALLING SEQUENCE
16              ;                                                       C?ATF
17              ;            P?REM ARG 1, ARG 2, ARG 3, ARG 4, ARG 5, C?ATP, ARG 7
18              ;                                                       BLANK
19              ;
20              ;5.1.1.2.1      ARGUMENTS
21              ;
22              ;                       ARG 1   IS THE LOCATION WHERE THE PHYSI-
23              ;                               CAL MEMORY SIZE WILL BE STORED.
24              ;                               (OR THE HIGHEST 1K BLOCK IF MAPPED)
25              ;                       ARG 2   IS THE LOCATION WHERE THE TOP-
26              ;                               MOST MEMORY LOCATION FOR THE
27              ;                               DIAGNOSTIC PROGRAM WILL BE
28              ;                               STORED.
29              ;                       ARG 3   IS THE LOCATION WHERE THE FIRST
30              ;                               STATUS WORD WILL BE STORED.
31              ;                       ARG 4   IS THE LOCATION WHERE THE SECOND
32              ;                               STATUS WORD WILL BE STORED
33              ;                       ARG 5   IS THE STARTING ADDRESS OF THE
34              ;                               PRE?M MACRO. (SEE NOTE).
35              ;                       ARG 6
36              ;                       C?ATF   (OPTIONAL) THESE CALLS MUST BE
37              ;                       C?ATP   USED IF THE USER WANTS THE
38              ;                               CAT/KITTEN PROGRAM TO RUN.
39              ;                               C?ATF MASKS OUT THE TTI, TTO,
40              ;                               LPT, AND MTA INTERRUPTS.
41              ;                               C?ATP MASKS OUT ALL OF THE
42              ;                               ABOVE, EXCEPT THE TTI.
43              ;                       ARG 7   CPU NUMBER TO INHIBIT EXPANSION
44              ;                               OF UNNECESSARY CODE WHEN THE CPU
45              ;                               TYPE IS PRE-DEFINED. BY DEFAULT, (NO
46              ;                               ARGUMENT PASSED), ALL CODE IS
47              ;                               EXPANDED. THE NUMBERS ARE
48              ;
49              ;                               AS FOLLOWS:
50              ;                                       0 = 1200
51              ;                                       1 = 800
52              ;                                       2 = NOVA/2
53              ;                                       3 = NOVA/3
54              ;                                       4 = MICRONOVA
55              ;                                       5 = ECLIPSE
56              ;                                       6 = RESERVED
57              ;                                       7 = RESERVED
58              ;
59              ;
60              ;5.1.1.2.2      EXAMPLE OF A MACRO CALL FOR ECLIPSE
```



```
01              ;               P?REM MTOP,TTOP,STAT1,STAT2,BEGIN,C?ATF,5
02              ;
03              ; NOTE: IF THE USER INTENDS TO USE THE PAGE ZERO DEFINITION
04              ;       MACROS P?GOO OR M?PG0, AND HE ALSO INTENDS TO START
05              ;       HIS DIAGNOSTIC PROGRAM BY CALLING THE PRE?M MACRO,
06              ;       THEN HE SHOULD PASS THE LABEL "BEGIN" AS THE FIFTH
07              ;       ARGUMENT OF THE CALL.
08              ;       IF THE USER INTENDS TO USE THE P?GOU MACRO AND
09              ;       TO START HIS PROGRAM BY CALLING THE PRE?M MACRO,
10              ;       HE SHOULD PASS THE SAME STARTING ADDRESS TO THE
11              ;       PRE?M MACRO AS TO THE P?GOU MACRO.
12              ;       IF THE USER DOES NOT INTEND TO USE ANY OF THE PAGE
13              ;       ZERO DEFINITIONS MACROS, HE CAN PASS ANY STARTING
14              ;       ADDRESS.
15              ;
16              ;NOTE:  IF SIZING A SYSTEM WITH AN "MMPU1", THE BANK JUMPERS
17              ;       MUST BE PROPERLY INSERTED OR INCORRECT SIZING WILL
18              ;       RESULT.
19              ;
20              ;5.1.1.3        SIZING
21              ;
22              ;       SIZES LOGICAL AND PHYSICAL MEMORY
23              ;       CHECKS IF CAT/KITTEN PROGRAM WAS LOADED
24              ;       IF CAT/KITTEN WAS LOADED IT PROVIDES ROOM AT MEMORY
25              ;       TOP FOR MINI-MONITOR, I/O MODULE, CAT/KITTEN PROGRAM
26              ;       AND THE BUFFERS FOR THE DISK/IORD OPERATIONS
27              ;       IF CAT/KITTEN WAS NOT LOADED IT PROVIDES ROOM ONLY
28              ;       FOR MINI-MONITOR AND I/O MODULE
29              ;
30              ;       CHECKS FOR THE PRESENCE OF A MAP AND THE MAP TYPE
31              ;       WHERE APPLICABLE. PROVIDES THE HIGHEST 1K PHYSICAL
32              ;       BLOCK OF MEMORY (IN OCTAL)
33              ;
34              ;5.1.1.4        THE MACRO ALSO SETS TWO STATUS WORDS WHICH CONTAIN
35              ;       THE FOLLOWING:
36              ;
37              ;               (STATUS WORD 1)
38              ;
39              ;       BIT NUMBER      SIGNIFICANCE
40              ;
41              ;       0 - 7           SPARE
42              ;       8 - 10          RESERVED FOR I/O MODULE OP
43              ;                       CODES (NOT IMPLEMENTED)
44              ;       11              0 = NO CAT/KITTEN
45              ;                       1 = CAT/KITTEN PROG HAS BEEN LOADED
46              ;       12              0 = MANUAL OR DEBUG MODE
47              ;                       1 = AUTO OR SEMI-AUTO MODE
48              ;       13 - 15         # OF PASSES OF PROG TO RUN
49              ;
50              ;
51              ;               (STATUS WORD 2)
52              ;
53              ;       BIT NUMBER      SIGNIFICANCE
54              ;
55              ;       0               0 = UNMAPPED
56              ;                       1 = MAPPED
57              ;
58              ;       1               0 = MMPU (ECLIPSE ONLY)
59              ;                       1 = MMPU1 (ECLIPSE ONLY)
60              ;
```

```
;       2 - 12          SPARE
;
;       13 - 15         0 = 1200 NOVA/2 A00
;                       1 = RESERVED
;                       2 = RESERVED
;                       3 = NOVA/3
;                       4 = MICRONOVA
;                       5 = ECLIPSE
;                       6 = RESERVED
;                       7 = RESERVED
;
;5.1.1.5        THE MACRO ALSO PROVIDES A POINTER TO THE
;       I/O MODULE. THE CALLING SEQUENCE IS:
;
;                       JSR @IOM?OD     ; P
;                       OP CODE
;
;5.1.1.6        THE MACRO ALSO PROVIDES A POINTER TO THE START
;       OF THE CAT/KITTEN PROGRAM (IF THE USER PASSES
;       THE SIXTH ARGUMENT OF THE MACRO CALL)
;       THE CALLING SEQUENCE IS:
;                       JSR   @ICA?T
;
;5.1.1.7        PAGE ZERO DEFINITION REQUIREMENTS
;
;       1)      THE FIRST FOUR ARGUMENTS USED IN THE MACRO
;               CALL MUST BE DEFINED IN PAGE ZERO
;
;       2)      THE POINTER TO  THE I/O MODULE MUST BE
;               DEFINED IN PAGE ZERO AS FOLLOWS:
;                       IOM?OD:  0
;
;       3)      IF THE USER WANTS THE CAT/KITTEN PROGRAM TO RUN
;               HE  MUST DEFINE IN PAGE ZERO THE FOLLOWING:
;                       ICA?T:  0
;
;       4)      IF THE USER HAS NOT USED EITHER THE DEFAULT
;               PAGE ZERO DEFINITIONS MACRO (P?G0D) OR THE
;               USER DEFINED PAGE ZERO DEFINITIONS MACRO
;               (P?G0U) HE MUST DEFINE A POINTER TO THE EGGS
;               BLOCK AS FOLLOWS:
;                       IEGG?S:  EGGS
;
;       5)      IF THE USER DOES NOT INTEND TO USE ANY OF THE
;               PAGE ZERO DEFINITIONS MACROS, HE MUST DEFINE
;               THE FOLLOWING PAGE ZERO COUNTER:
;                       PAS?S:  0
;       P?STD   5.1.2
;
;5.1.2  P?STM
;
;       THIS MACRO IS USED TO HANDLE ALL POST-DIAGNOSTIC
;       PROGRAM COMMUNICATIONS WITH DTOS
;       IT'S PURPOSE IS TO MONITOR THE EGGS BLOCK AND THE DTOS
;       PASS COUNTERS (PA?SIN AND PAS?S)
;       IT WILL DECIDE WHETHER TO RETURN TO THE START OF THE
;       DIAGNOSTIC PROGRAM, JUMP TO A USER DEFINED PRINT MACRO
;       (OR PRINT ROUTINE) OR TO START THE CAT/KITTEN PROGRAM
;
;       THE USER MAY DEFINE FOUR ARGUMENTS IN THE MACRO
```



```
;       CALL. THE FIRST ARGUMENT MUST BE DEFINED, THE
;       NEXT TWO ARE OPTIONAL. THESE ARE:
;
;       ARG 1   THE "START OF A NEW PASS" ADDRESS
;
;       ARG 2   (OPTIONAL) THE USER CAN EITHER PASS THE MACRO
;       P?PAS   CALL "P?PAS" AS THE SECOND ARGUMENT, IN WHICH
;               CASE, THE FOLLOWING MESSAGE WILL BE PRINTED:
;
;                       " PASS  # "
;               WHERE # IS THE PRESENT PASS COUNT IN DECIMAL
;
;               OR HE COULD PASS A CALL TO A USER DEFINED
;               PRINT MACRO OR A CALL WHICH EXPANDS TO A JSR
;               TO A PRINT ROUTINE. (IF THE USER DOES NOT
;               INTEND TO USE THE SECOND ARGUMENT (NO PRINT-
;               OUT DESIRED) HOWEVER HE DOES INTEND TO USE
;               THE THIRD ARGUMENT (L?CAT) HE SHOULD PLACE TWO
;               CONSECUTIVE COMMAS DIRECTLY AFTER ARGUMENT 1).
;
;       ARG 3   (OPTIONAL) IF THE USER WANTS THE CAT/KITTEN
;       L?CAT   PROGRAM TO RUN HE MUST PASS "L?CAT" AS THE
;               THIRD ARGUMENT OF THE MACRO CALL.
;
;       ARG 4   (OPTIONAL) THIS ARGUMENT MAY BE USED TO INSERT
;               USER DEFINED CODE.
;
; THE CALLING SEQUENCE IS:
;                               P?STM ARG 1,ARG 2,L?CAT,ARG 4
;                                       P?PAS
;
;               WHERE   ARG 1   IS THE STARTING ADDRESS OF A NEW
;                               PASS
;
;                       ARG 2   (OPTIONAL) THE USER CAN EITHER
;                       P?PAS   PASS THE MACRO CALL"P?PAS" AS
;                               SECOND ARGUMENT, IN WHICH CASE
;                               THE FOLLOWING MESSAGE WILL BE
;                               PRINTED:
;                                               " PASS  # "
;                               WHERE # IS THE PRESENT PASS COUNT
;                               IN DECIMAL
;
;                               OR THE USER COULD PASS A CALL TO
;                               A USER DEFINED PRINT MACRO OR A
;                               CALL WHICH EXPANDS TO A JSR TO A
;                               USER DEFINED PRINT ROUTINE.
;                               (IF THE USER DOES NOT INTEND TO
;                               USE THE SECOND ARGUMENT (NO PRINT-
;                               OUT DESIRED) HOWEVER HE DOES INTEND
;                               TO USE THE THIRD ARGUMENT (L?CAT)
;                               HE SHOULD PLACE TWO CONSECUTIVE
;                               COMMAS DIRECTLY AFTER ARGUMENT 1)
;
;                       L?CAT   (OPTIONAL) IS THE CALL WHICH THE
;                               USER MUST SPECIFY IF HE WISHES
;                               THE CAT/KITTEN PROGRAM TO BE
;                               STARTED AFTER THE FIRST PASS, IF
;                               IT IS LOADED.
;
```

```
01              ;                       ARG 4   (OPTIONAL) IF USED, SHOULD BE A
02              ;                               MACRO CALL TO EXPAND USER DEFINED
03              ;                               CODE.
04              ;
05              ; EXAMPLE OF THE MACRO CALL:
06              ;
07              ;                               P?STM NUPASS,,L?CAT
08              ;
09              ; THIS CALL WILL CAUSE THE MACRO P?STM TO RETURN DIRECTLY
10              ; TO "NUPASS", THE LOCATION THE USER DEFINED AS THE
11              ; START OF THE NEXT PASS OF THE DIAGNOSTIC PROGRAM, IT WILL
12              ; NOT PRINT AFTER EACH PASS, BUT IT WILL RUN THE CAT/KITTEN
13              ; PROGRAM.
14              ;
15              ;
16              ; PAGE ZERO REQUIREMENTS:
17              ;
18              ;       IF THE USER INTENDS TO ALLOW THE CAT/KITTEN PROGRAM TO
19              ;       RUN, HE MUST DEFINE IN PAGE ZERO THE FOLLOWING:
20              ;                               ICA?T:  0
21              ;
22              ;       IF THE USER DOES NOT INTEND TO USE THE P?REM MACRO IN
23              ;       CONJUNCTION WITH THE P?STM MACRO, HE MUST SUPPLY THE
24              ;       STARTING ADDRESS OF THE CAT/KITTEN PROGRAM, THIS ADDRESS
25              ;       MUST BE STORED IN LOCATION "ICA?T".
26              ;
27              ;       THE PRINT MACRO "P?PAS" USES THE TTYIO PACKAGE, THEREFORE
28              ;       IF THE USER INTENDS TO USE THE "P?PAS" MACRO HE SHOULD
29              ;       CALL THE "T?TY0" MACRO. IF THE USER DOES NOT INTEND TO
30              ;       USE THE PAGE ZERO DEFINITIONS MACROS, HE MUST ALSO DEFINE
31              ;       THE  FOLLOWING PAGE ZERO POINTERS:
32              ;
33              ;                               IMES?S: MES?S
34              ;                               IPDE?C: PDE?C
35              ;
36              ;       IF THE USER DOES NOT INTEND TO USE EITHER THE DEFAULT
37              ;       PAGE ZERO DEFINITIONS MACRO (P?G0D) OR THE USER
38              ;       DEFINED PAGE ZERO DEFINITIONS MACRO (P?G0U), HE MUST
39              ;       DEFINE THE FOLLOWING IN PAGE ZERO:
40              ;
41              ;                               IEGG?S: EGGS
42              ;                               PAS?S:  0 **
43              ;                               PA?SIN: *
44              ;                               PA?SVL: *
45              ;
46              ;       * -THESE CONSTANTS MUST BE DEFINED WITH SOME VALUE
47              ;          OTHER THAN ZERO, IF THE USER DOES NOT INTEND TO
48              ;          USE THE INTERNAL PASS COUNTERS SET THEM TO "1".
49              ;
50              ;       ** ONLY THIS COUNTER IS DEFINED BY THE M?PG0 MACRO.
51              ;
52              ; PROGRAM SEQUENCE:
53              ;
54              ;       THE MACRO P?STM DOES THE FOLLOWING:
55              ;
56              ;       FIRST, IT WILL CHECK IF A PASS HAS BEEN COMPLETED, THIS
57              ;       OCCURS WHEN PA?SIN=0, AT THIS POINT IT WILL EITHER EXE-
58              ;       CUTE THE CALL SPECIFIED BY ARGUMENT 2 OR CONTINUE IT'S
59              ;       OWN SEQUENCE IF ARGUMENT 2 HAS NOT BEEN SPECIFIED.
60              ;
```



```
01              ;       NEXT THE MACRO WILL CHECK IF IT HAS JUST COMPLETED THE
02              ;       FIRST PASS OF THE DIAGNOSTIC PROGRAM, IF SO IT WILL
03              ;       CHECK WHETHER THE CAT/KITTEN HAS BEEN LOADED BY DTOS, IF
04              ;       IT HAS IT WILL START THE CAT/KITTEN.
05              ;
06              ;       LASTLY, IT WILL CHECK THE MODE OF OPERATION, IF IT IS IN
07              ;       MANUAL MODE OR AUTO MODE WITH THE PASS COUNT IN THE EGGS
08              ;       BLOCK NOT EQUAL TO 0, THE MACRO WILL RETURN TO THE START
09              ;       OF THE NEXT PASS OF THE DIAGNOSTIC PROGRAM, HOWEVER IN
10              ;       AUTO MODE WITH THE PASS COUNT EQUAL TO 0 THE PROGRAM
11              ;       WILL RETURN TO DTOS.
```

```
10015 N4MPD
01              ;       5.2     MONITOR LOCATIONS
02              ;
03              ;               LOC 200         USED BY DTOS
04              ;               LOC 201         CURRENT TEST ADDRESS LOCATION
05              ;               LOC 202         STARTING ADDRESS FOR  PROGRAM
06              ;               LOC 203         PROGRAM PASS COUNT
07              ;               LOC 204         TEST ITERATION COUNT
08              ;
09              ;       5.3     MEMORY MAP
10              ;
11              ;               UNLESS OTHERWISE MENTIONED IN THE
12              ;               SPECIFIC TEST,MEMORY FOR USERS A
13              ;               AND B IS ALWAYS MAPPED TO ITSELF.
14              ;               THAT IS LOGICAL=PHYSICAL.
15              ;
16              ;
17              ;6.0    SEQUENCE OF TESTING
18              ;
19              ;               THE TESTING SEQUENCE FOR PROGRAM
20              ;               N3MAP IS AS FOLLOWS:
21              ;
22              ;               LABEL           DESCRIPTION
23              ;               -------------   ------------------------------
24              ;
25              ;               A SERIES        UNMAPPED LOGIC TESTS
26              ;               A00 - A06       DIAGNOSTIC RESET TESTS
27              ;               A07 - A15       STATUS REGISTER TESTS
28              ;               A14 -           VIOLATION DATA REG. TEST
29              ;               A15 -           VIOLATION PC REGISTER TEST
30              ;               A19             INVALID SINGLE CYCLE TEST
31              ;               A20 - A27       MAP ENTRY REG. TESTS
32              ;               A28             PGM MAP INHIBIT-TRAP TEST
33              ;               A29 - A31       MAPPED SINGLE CYCLE TESTS
34              ;
35              ;               B SERIES        VIOLATION CIRCUITRY TESTS
36              ;               B00 - B03       WRITE VIOL. TESTS
37              ;               B04 -           I/O VIOL. TEST
38              ;               B05 - B06       AUTO INDEX VIOL. TESTS
39              ;               B07 - B08       VALIDITY PROTECT VIOL. TESTS
40              ;               B09             WRITE PROTECT(USER B)
41              ;               B10 - B11       VALIDITY VIOL.(USER B)
42              ;               B12             VIOL. PC REGISTER TEST
43              ;
44              ;               C SERIES        PROGRAM MAP ENABLE TESTS
45              ;               C00             PGM MAP ENTRY
46              ;               C01             MAP DONE FLAG TEST
47              ;               C02 - C04       DEFER VIOL. TESTS
48              ;               C06             I/O VIOLATION TEST
49              ;               C07             WRITE PROTECT VIOL.(USER A)
50              ;               C08 - C10       PAGE CHANGING,STATUS BIT CHANGING,
51              ;                               MAP ENTRY CHANGING WHILE MAPPING
52              ;               C11             PROGRAM MAP INHIBIT TEST
53              ;
54              ;               D SERIES        I/O TESTER DCH TESTS
55              ;               D00 - D02       DCH  TESTS
56              ;               D03 - D06       DCH MAPPING TESTS
57              ;
58              ;               E SERIES        FPU TESTS
59              ;               E00             FPU DCH TEST
60              ;               E01             FPU WHILE MAPPING TEST
```

```
0016 N4MPD
01              ;               E02             FPU USING PGM MAP TEST
02              ;               E03             FPU DCH ERROR
03              ;               E04             FPU,I/O TESTER INTERREACTION TEST
```

```
10017 N4MPD
01              ;
02              ;7.0    PROGRAMMING DESCRIPTION FOR MMPU FEATURE
03              ;
04              ;               THE INSTRUCTIONS WHICH MAY BE
05              ;               USED TO SETUP AND INTERROGATE
06              ;               THE MMPU FEATURE ARE BRIEFLY
07              ;               DESCRIBED HERE.
08              ;
09              ;               7.1     READ STATUS COMMMAND (DIA AC,2)
10              ;
11              ;               7.2     WRITE STATUS COMMAND (DOA AC,2)
12              ;
13              ;
14              ;                       THE STATUS BITS SIGNIFICANCE
15              ;                       ARE AS FOLLOWS:
16              ;
17              ;                       BIT     CONTENTS
18              ;
19              ;                       0       PGM MAPPING ENABLED
20              ;                       1       DCH MAPPING ENABLED
21              ;                       2       PROGRAM MAP INHIBIT(READ ONLY)
22              ;                       3-8     UNUSED(0'S)
23              ;                       9       SINGLE CYCLE WRITE PROTECT ENABLE
24              ;                       10      SINGLE CYCLE PROGRAM MAP A/B
25              ;                       11      AUTO INDEX PROTECT
26              ;                       12      DEFER PROTECT ENABLE
27              ;                       13      I/O PROTECT ENABLE
28              ;                       14      WRITE PROTECT ENABLE
29              ;                       15      PROGRAM MAP A/B
30              ;
31              ;
32              ;               7.3     WRITE MAP ENTRY COMMAND (DOB AC,2)
33              ;
34              ;               7.4     READ MAP ENTRY COMMAND (DIA AC,3)
35              ;
36              ;
37              ;                       THE SIGNIFICANCE OF THE MAP ENTRY BITS
38              ;                       ARE AS FOLLOWS:
39              ;                       BIT     CONTENTS
40              ;
41              ;                       0       PROGRAM/DATA CHANNEL MAP
42              ;                       1-5     PAGE
43              ;                       6       USER SELECTION  0=A, 1=B
44              ;                       7       WRITE PROTECT
45              ;                       8-15    EXTENDED ADDRESS
46              ;
47              ;
```

```
10018 N4MPD
01              ;
02              ;
03              ;               7.5     READ VIOLATION PC COMMAND (DIB AC,3)
04              ;
05              ;
06              ;                       THE BIT SIGNIFICANCE IS AS FOLLOWS:
07              ;
08              ;                       BIT     CONTENTS
09              ;
10              ;                       0       NOT USED(WILL BE 0)
11              ;                       1-15    SAVED PC BITS 1-15
12              ;
13              ;
14              ;
15              ;               7.6     READ VIOLATION DATA COMMAND (DIB AC,2)
16              ;
17              ;                       THE BIT SIGNIFICANCE IS AS FOLLOWS:
18              ;
19              ;                       BIT     CONTENTS
20              ;
21              ;                       0       VIOLATION FLAG
22              ;                       1-5     VIOLATION PAGE
23              ;                       6-8     NOT USED(WILL BE 0'S)
24              ;                       9       SSE ERROR
25              ;                       10      VALIDITY PROTECTION ERROR
26              ;                       11      AUTO INDEX PROTECTION ERROR
27              ;                       12      DEFER PROTECTION ERROR
28              ;                       13      I/O PROTECTION ERROR
29              ;                       14      WRITE PROTECTION ERROR
30              ;                       15      VIOLATION A/B USER
31              ;
32              ;
33              ;               7.7     SELECT PAGE COMMAND (DOA AC,3)
34              ;
35              ;                       BIT     CONTENTS
36              ;
37              ;                       0       PROGRAM/DCH MAP
38              ;                       1-5     PAGE
39              ;                       6       USER SELECTION A/B
40              ;                       7-15    UNUSED BITS
41              ;
```

```
10019 N4MPD
01              ;
02              ;
03              ;4.0    I/O TESTER HARDWARE DESCRIPTION
04              ;
05              ;       4.1     TEST BOARD COMMANDS
06              ;
07              ;               IORST  -  CLEAR THE TESTER
08              ;               NIOC 0 -  CLEAR THE TESTER (IN  NEW MODE)
09              ;               INTA   -  READ THE DATA BUFFER (NOT NEW MODE)
10              ;               DATIC  -  READ THE PULSE DETECTORS
11              ;               DATIB  -  READ THE DATA BUFFER
12              ;               DATIA  -  READ THE DCH ADDRESS BUFFER (NEW MODE)
13              ;               DATOA  -  LOAD THE DATA BUFFER
14              ;               DATOB  -  LOAD THE FUNCTION BUFFER
15              ;               DATOC  -  LOAD THE DATA AND DCH ADDRESS BUFFERS
16              ;
17              ;       4.2     FUNCTION REGISTER BIT ASSIGNMENTS
18              ;
19              ;               BIT 0      SET DCH SYNC
20              ;               BIT 1      SET DCH MODE0
21              ;               BIT 2      SET DCH MODE1
22              ;               BIT 3      SET PI SYNC
23              ;               BIT 4      BUSY (IF NOT IN NEW MODE)
24              ;               BIT 5      DONE (IF NOT IN NEW MODE)
25              ;               BIT 6      NEW MODE
26              ;               BITS 7-9   AN OCTAL # WHICH SPECIFIES THE
27              ;                          # OF RQENB PULSES BETWEEN
28              ;                          SUCCESSIVE DCH CYCLES. (NEW MODE ONLY)
29              ;                          NOTE THAT 0 SPECIFIES 1 RQENB PULSE.
30              ;               BITS 10-15  # OF DCH CYCLES TO BE RUN.
31              ;                          (NEW MODE ONLY)
32              ;                          NOTE THAT 0 SPECIFIES 1 DCH CYCLE.
33              ;
34              ;
```

```
10020 N4MPD
01
02              ;
03              ;       4.3     PULSE DETECTOR BIT ASSIGNMENTS
04              ;
05              ;               BIT 0      IOPLS
06              ;               BIT 1      INTA (INTA AND DCHP)
07              ;               BIT 2      MSKO
08              ;               BIT 3      DCMI
09              ;               BIT 4      OVFLO
10              ;               BIT 5      DCMO
11              ;               BIT 6      DCMA
12              ;               BIT 7      RQENB (COMPLEMENTS WITH EACH PULSE)
13              ;               BIT 8      DATOA
14              ;               BIT 9      DATOB
15              ;               BIT 10     DATOC
16              ;               BIT 11     DATIA
17              ;               BIT 12     DATIB
18              ;               BIT 13     DATIC (NOT SET IF DEV CODE = 0)
19              ;               BIT 14     STRT
20              ;               BIT 15     CLR
21              ;
22              ;       4.4     TEST BOARD LOGIC
23              ;
24              ;               THE TEST BOARD CONTAINS 16 PULSE
25              ;               DETECTOR FLIP FLOPS.  THESE FF'S MAY
26              ;               BE READ BY A "DIC" WITH A DEVICE
27              ;               CODE OF 0.  THEY MAY BE CLEARED BY
28              ;               IORST OR NIOC 0 (IF IN NEW MODE).
29              ;               A PARTICULAR FF SETS WHENEVER
30              ;               A PULSE OCCURS ON THE LINE TO
31              ;               WHICH IT IS CONNECTED.
32              ;
33              ;               THE TEST BOARD ALSO CONTAINS
34              ;               A 16 BIT DATA BUFFER.  THIS
35              ;               BUFFER MAY BE LOADED/READ ETC.
36              ;               UNDER PROGRAM CONTROL.  THIS
37              ;               BUFFER IS ALSO USED FOR DCH
38              ;               OPERATIONS.  IT SHOULD BE NOTED THAT
39              ;               IN NEW MODE, ANY LOAD DATA BUFFER
40              ;               PROCEDURE, ACTUALLY LOADS THE
41              ;               EXCLUSIVE OR OF THE OUTPUT DATA
42              ;               AND THE DATA PREVIOUSLY STORED IN THE BUFFER.
43              ;
44              ;
45              ;               A 15 BIT DCH ADDRESS BUFFER
46              ;               IS USED TO DIRECT DCH REQUESTS
47
48
49              .EOT
```

```
0001  24M04    MACRO REV 06.50          14:58:30 12/24/78
01
02
03
04
05
06
07              ;*****************************************************************
08              ;
09              ; NAME: 4MMOM.TX                          PART NUMBER: 097-001132
10              ;
11              ;
12              ; DESCRIPTION: NOVA 4 MEMORY DIAGNOSTIC / MAPPED
13              ;
14              ;
15              ; REVISION HISTORY:
16              ;
17              ;       REV.            DATE
18              ;
19              ;       00              12/29/78
20              ;
21              ;
22              ; COPYRIGHT (C) DATA GENERAL CORPORATION, 1978
23              ; ALL RIGHTS RESERVED.
24              ;*****************************************************************
```

24MOM

```
10002 NAMOM
01          000000          .DO     0
02                                          CONTENTS
03                                          ========
04
05
06              I. GOAL
07
08              II. DESIRED ATTRIBUTES
09
10              III. WHAT THE PROGRAM WILL DO AND WHY
11
12              IV. DESIGN HIGHLIGHTS
13
14              V. KNOWN ANOMALIES
15
16
17              TABLE 1- TEST NAMES' BIT POSITIONS
18
19              TABLE 2- CONTROL FLAGS' DEFAULT SETTINGS
20
21              TABLE 3- SMALLEST AREA OF MEMORY THAT CAN BE TESTED,
22                       LISTED BY TEST NAME
23
24              TABLE 4- PROGRAM MONITOR COMMANDS
25
26
27              FIGURE 1- FAULT TYPES VS. MEMORY TEST PATTERNS
28
29              FIGURE 2- ERROR LOG OUTPUT
30
31              FIGURE 3- MEMORY TOPOLOGY, 4K RAMS
32
33              FIGURE 4- MEMORY TOPOLOGY, 16K RAMS
34
35
36
37                      LAST MODIFICATION:  12/27/78   14:38
```

I. GOAL
====

PROVIDE A MEANS OF TESTING UP TO 256K WORDS OF MOS DYNAMIC RAMS ON THE NOVA 4.

II. DESIRED ATTRIBUTES
==========

A) SPEED- ENTIRE PROGRAM SHOULD RUN NO LONGER THAN 10-15 MINUTES IN AUTO MODE.

B) THOROUGHNESS- ALL MEMORY FAULT TYPES SHOULD BE TESTED.

C) STRENGTH- EACH MEMORY FAULT TYPE SHOULD BE CHECK BY A MEMORY TEST PATTERN OF AT LEAST MEDIUM STRENGTH. THE STRONGER A TEST, THE MORE LIKELY IT IS TO FIND A BUG. IN A SIMILAR VEIN, "WEAK" TESTS FIND HARD ERRORS, "STRONG" TESTS FIND SOFT (I.E., INTERMITTENT, INTERACTIVE) ERRORS.

D) DTOS COMPATIBLE.

E) BE USED UNDER ANY SIZE MEMORY (UP TO 256K WORDS) USING ANY OF THE FOLLOWING BOARD TYPES:

1. 16K WITH 4K RAMS
2. 32K " 4K "
3. 64K " 16K "
4. 128K " 16K "

F) OPTIONS- UNDER MANUAL CONTROL, THE USER MAY OPT FOR TESTS WHICH COVER LESS IMPORTANT FAULT TYPES AND/OR TAKE A LONG TIME TO RUN (GREATER THAN 1 HOUR). HE CAN ALSO ENABLE SPECIAL FEATURES WHICH AID IN THE LOCATION OF DIFFICULT ERRORS.

G) IF AN ERROR IS FOUND, PRINT OUT THE EXACT LOCATION OF THE BAD CHIP.

H) PROGRAM LENGTH WILL BE 8K OR LESS. THIS ALLOWS THE TESTING OF 4K RAMS/16K BOARD SYSTEMS.

I) AN ERROR LOG BE KEPT, WHICH CONTAINS A HISTORY OF ALL ERRORS ENCOUNTERED.

III. WHAT THE PROGRAM WILL DO AND WHY
==========

A) IN AUTO MODE

DEFINITIONS IN AUTO MODE, ALL CONTROL FLAGS (CF'S) ARE SET TO DEFAULT. SEE TABLE 2 FOR DEFAULT SETTINGS. 'AUTO MODE' IS RUN BY ISSUING ANY AUTO MODE COMMAND IN DTOS (E.G., LOAD.NN MEMD, AUTO). ALSO, AUTO MODE CAN BE SIMULATED BY RUNNING UNDER DTOS MANUAL MODE BUT USING THE DEFAULT SETTINGS FOR

ALL CF'S.

1. SIZE MEMORY- RUN A TOP DOWN MEMORY SIZING ROUTINE TO DETERMINE THE TOP OF MEMORY.

STARTING WITH THE FIRST BOARD, DETERMINE WHAT CONFIGURATION IT IS. THE FOUR POSSIBLE BOARD CONFIGURATIONS ARE LISTED IN II.E). THIS IS NECESSARY FOR CONTROL OF VARIOUS PARAMETERS WITHIN THE PROGRAM, ESPECIALLY IN DETERMINING THE EXACT LOCATION OF ANY ERRORS (BOARD, MODULE, BANK, BIT).

2. TEST THE BOARD

SEVEN TESTS HAVE BEEN CHOSEN THAT MOST OPTIMALLY MEET THE DESIRED ATTRIBUTES: 1) MODIFIED DATA=ADDRESS; 2) DATA=ADDRESS UPPER; 3) STUCK ADDRESS BIT; 4) MARCHING 1/0; 5) GALLOPING COLUMN; 6) GALLOPING ROWS; AND 6) GALLOPING DIAGONAL.

THE TESTS WILL BE RUN IN THAT ORDER. THIS WAY, IF THE FAULT IS A HARD ERROR, THEN ONE THE FIRST THREE QUICK TESTS WILL FIND IT. IF NOT, THE GALLOPING TESTS WILL THEN BE RUN. SEE TABLE 1 FOR APPROXIMATE TEST TIMES FOR EACH PATTERN.

THE MODIFIED DATA=ADDRESS TEST WILL CONSIST OF TWO PHASES. IN THE FIRST PHASE OF THE MODIFIED DATA=ADDRESS TEST, THE DATA WORD WRITTEN TO AND SUBSEQUENTLY READ FROM THE MEMORY LOCATION WILL BE THE LOWER 16 BITS OF ITS 20-BIT ADDRESS. ON THE SECOND PHASE, THE COMPLEMENT OF THE LOWER 16 BITS OF THE 20-BIT ADDRESS WILL BE WRITTEN TO/READ FROM THE LOCATIONS.

THE DATA=ADDRESS UPPER IS ESSENTIALLY THE THIRD PHASE IN THE D=A TESTING. THE D=A UPPER TEST WILL WRITE AND SUBSEQUENTLY READ THE UPPER 4 BITS OF THE PHYSICAL ADDRESS OF THE LOCATION UNDER TEST.

THIS METHOD CHECKS FOR CELL FUNCTIONALITY (FIRST AND SECOND PHASE), ADDRESSING ERRORS BETWEEN ALL 1K PHYSICAL BLOCKS (THIRD PHASE), AND ADDRESSING ERRORS WITHIN EACH 16K PHYSICAL BLOCK (FIRST PHASE). PHASE 1 AND 2 ARE RUN IN 16K 'CHUNKS', AND PHASE 3 IS RUN IN 1K "CHUNKS".

THE GALLOPING TEST PATTERNS WILL BE RUN IN 16K CHUNKS FOR 4K RAMS AND IN 32K CHUNKS FOR 16K RAMS. IN EACH CASE, EXTENSIVE USE OF THE MMPU1 AND MICROCODE SUPPORT IS MADE.

THESE SEVEN TESTS COVER THE FOLLOWING SEVEN FAULT TYPES: 1) FAULTY ADDRESS DECODERS; 2) INTERACTIVE COUPLING BETWEEN CELLS; 3) REFRESH SENSITIVITY; 4) SLOW ACCESS TIME; 5) SENSE AMPLIFIER RECOVERY; 6) CELL FUNCTIONALITY (CAN THE CELL HOLD WHAT IT'S TOLD TO); AND 7) STUCK ADDRESS BITS AT THE CHIP LEVEL.

FIGURE 1 DETAILS WHICH TESTS COVER WHAT FAULT TYPES.

EACH TEST WILL BE RUN SEQUENTIALLY DEPENDENT ON ERROR CONDITIONS. SPECIFICALLY, TEST 1 IS RUN. IF AN ERROR OCCURS, A PRINTOUT EXPLAINING THE NATURE AND LOCATION (BOARD, MODULE, BANK, AND BIT) OF THE ERROR IS GENERATED AND CONTROL IS RETURNED TO DTOS. IF NO ERROR OCCURS, TEST 2 IS RUN. EACH TEST IS RUN IN A SIMILAR MANNER UNTIL AN ERROR OCCURS OR ALL THE TESTS ARE COMPLETED. A COMPLETION MESSAGE IS THEN PRINTED OUT (SEE SECTION III. C) 2.).

FOR THE FIRST BOARD, TESTING PROCEEDS IN A SPECIALIZED MANNER (SEE IV. D). FOR THE SECOND THROUGH LAST BOARD, ALL THE SELECTED TESTS ARE RUN FIRST AGAINST BANK 0, AND THEN AGAINST BANK 1 (THIS ACCOUNTS FOR A DOUBLE LISTING OF TEST NAMES).

3. ASSUMING NO ERROR HAS OCCURRED ON THE PRESENT BOARD UNDER TEST, THE PROGRAM WOULD CONTINUE ON TO THE NEXT BOARD.

4. STEPS 1-3 ARE REPEATED FOR EACH BOARD IN MEMORY.

B) IN MANUAL MODE

DEFINITION: IN MANUAL MODE, THE OPERATOR CAN TAILOR THE PROGRAM TO EXAMINE MORE SPECIFIC PROBLEMS. THE OPERATOR MUST ENTER THE SETTINGS FOR ALL THE CF'S BEFORE THE TESTS CAN BE RUN (I.E., AT THE START OF THE FIRST PASS). A <LF> RESPONSE WILL SET THE CF TO ITS DEFAULT. SEE TABLE 2 FOR DEFAULT SETTINGS. MANUAL MODE IS RUN BY USING ANY DTOS MANUAL MODE COMMAND (E.G., LOAD MEMO, DEBUG MEMO).

ALL NUMERIC QUESTIONS SHOULD BE ANSWERED IN OCTAL NUMBERS. ALL YES/NO QUESTIONS MUST BE ANSWERED WITH "Y" <CR>, "N" <CR>, OR <LF>. THE <LF> SETS THE CF TO THE DEFAULT, WHICH IS 'NO'.

1. STARTING ADDRESS AND ENDING ADDRESS

BY ENTERING A STARTING ADDRESS (STADR) AND ENDING ADDRESS (ENDADR), THE USER ENABLES THE PROGRAM TO RUN SLOWER TESTS AND STILL COMPLETE THEM IN AN ACCEPTABLE AMOUNT OF TIME BY REDUCING THE SIZE OF MEMORY BEING TESTED.

THE STADR AND ENDADR ENTERED IS THE PHYSICAL 1K BLOCK OF THE INTENDED ADDRESSES. SINCE 1K=2000 OCTAL LOCATIONS, TO TEST PHYSICAL LOCATIONS 200000 TO 377777, FOR EXAMPLE, ENTER "100" AND "177" FOR STADR AND ENDADR, RESPECTIVELY.

RESTRICTIONS: 1) 0<=STADR<TOP ; AND 2) 0<ENDADR<=TOP. IF THE ENDADR ENTERED EXCEEDS TOP, THEN ENDADR IS SET TO TOP (OF MEMORY).

THE USER SHOULD NOTE THE SMALLEST AMOUNT OF MEMORY THAT CAN BE TESTED, BY TEST NAME, AS OUTLINED IN TABLE 3. FAILURE TO DO SO COULD RESULT IN CRASHING THE PROGRAM (I.E., IF USING GALLOPING PATTERNS, DON'T RELOCATE THE PROGRAM IN THE SAME BANK).

2. MODULES TO BE TESTED

ENTER A BIT PATTERN TO DETERMINE WHAT MODULES ARE TO BE TESTED. A "0" RESPONSE TESTS ALL MODULES. THIS CP APPLIES TO ALL TESTS EXCEPT THE MODIFIED DATA=ADDRESS AND DATA=ADDRESS UPPER.

MUT	MODULE TESTED	(MUT=MODULES UNDER TEST)
15	A	
14	B	
13	C	
12	D	

AS USUAL, A BIT SET MEANS THAT THAT MODULE WILL BE TESTED.

3. ERROR CONTROL MODE (ECM)

"ECM (0-4)? " =THE NUMBER ENTERED WILL AFFECT THE ACTION TAKEN UPON HITTING AN ERROR:

ECM	ACTION TAKEN
0	PRINT AN ERROR REPORT AND RETURN TO DTOS.
1	PRINT AN ERROR REPORT AND CONTINUE.
2	PRINT AN ERROR REPORT AND HALT.
3	PRINT AN ERROR REPORT AND GO TO PROGRAM MONITOR.
4	ENTER ERROR IN ERROR LOG AND CONTINUE.

ECM=2 IS ILLEGAL WHEN USING POWER SUPPLY VOLTAGE MARGINING. (SEE #) 5.)

THE DEFAULT ECM IS 3.
IF "4" IS ENTERED, THE USER MUST THEN SET THE LOG PRINT (LP) CONTROL FLAG:
"PRINT ERROR LOG AT CONCLUSION (Y/N)? "

FOR FURTHER INFORMATION -- III.A)7., III.C)1., III.D) .

4. SUPPLEMENTAL ERROR INFORMATION

"SUPPLEMENTAL ERROR INFORMATION (Y/N)? "

IF YES, THEN THE FAILURE ADDRESS, EXPECTED DATA, AND ACTUAL DATA ARE PRINTED IN ADDITION TO THE (DEFAULT) ERROR REPORT FORMAT. SEE III.C) .

5. POWER SUPPLY VOLTAGES (PSV)

THE PROGRAM CAN EITHER RUN THE TESTS UNDER NOMINAL CONDITIONS OR UNDER MARGINAL CONDITIONS. TO RUN UNDER MARGINAL VOLTAGES, THE BACKPANEL PROGRAMMING PLUG MUST BE IN THE CORRECT POSITION.

THE USER ENTERS AN OCTAL BIT PATTERN AS FOLLOWS:

BIT SET	VAA (-5V)	VDD (+12V)	VCC (+5V)
15	-5.25	10.8	4.5
14	-4.5	12.6	5.25

IF NO BITS ARE SET, THE TESTS WILL RUN UNDER NOMINAL VOLTAGES (WHICH IS THE DEFAULT CONDITION). OTHERWISE, THE TESTS WIL BE RUN UNDER EACH CHOSEN MARGINAL CONDITION.

EXAMPLES (OCTAL NUMBERS): 1=HI/LO/LO ONLY; 2=LO/HI/HI ONLY; 3= FIRST RUN WITH HI/LO/LO, SECOND RUN WITH LO/HI/HI; 0= NOMINAL VOLTAGES ONLY.

PSV MAY NOT BE USED WHEN ECM=2 (HALT ON ERROR REPORT). THIS IS TO ASSURE THAT THE VOLTAGES ARE RETURNED TO NOMINAL LEVELS BEFORE ANY OTHER PROGRAM IS RUN. FOR THIS REASON, THE PROGRAM MONITOR'S "H" (HALT) COMMAND WILL BE REJECTED IF PSV>0.

PROGRAM USING SWITCH PACK), 'BREAK', OR HIT THE FRONT PANEL RESET SWITCH DURING ANY RUN WHERE VOLTAGES ARE BEING MARGINED. IF IT IS VITAL TO STOP THE PROGRAM DURING A PSV RUN, USE THE FOLLOWING SEQUENCE:

1) 'CNTRL-C': PUTS YOU IN THE PROGRAM MONITOR.
2) 'A': ABORTS THE PROGRAM, RETURNS VOLTAGES TO NOMINAL LEVELS, AND RETURNS TO DTOS.
3) 'EXIT': A DTOS COMMAND WHICH HALTS THE MACHINE.

IF YOU RESET PSV=0 USING THE PROGRAM MONITOR DURING A PSV RUN, THE PROGRAM WILL SUBSEQUENTLY ALLOW YOU TO USE THE "H" COMMAND. HOWEVER, THIS IS NOT ADVISABLE BECAUSE THIS WILL NOT RESET THE VOLTAGES TO THEIR NOMINAL VOLTAGES. AS A GENERAL RULE, WHEN USING PSV, AND YOU WANT TO STOP THE PROGRAM, USE THE PROCEDURE OUTLINED ABOVE.

AS USUAL, IF AT ANY TIME AN ERROR IS ENCOUNTERED, THE PROGRAM WILL PROCESS IT ACCORDING TO THE PRESENT ERROR CONTROL MODE.

6. ERCC ENABLE

"ENABLE ERCC (Y/N)? "

TO CHECK THE ERCC BITS, ANSWER "Y". WITH ERCC ENABLED, THE ERCC LOGIC WILL COPY THE ERCC BITS (MEMIN 16-20) INTO DATA BITS 11-15.

IF ERCC IS ENABLED AND AN ERROR IS REPORTED IN BITS 11-15, THEN THE USER SHOULD TRANSLATE THAT TO MEAN THE ERCC BITS (I.E., BITS 16-20). TO REMIND HIM OF THIS, A WARNING MESSAGE WILL BE PRINTED ALONG WITH THE ERROR PRINTOUT, AND IF LOGGING ERRORS, BIT 0 OF THE TEST NAMES' BIT PATTERN WILL BE SET.

NOTE THAT AN ECC ERROR IS TREATED JUST LIKE ANY OTHER ERROR WHICH WOULD BE DETECTED BY A TEST PATTERN.

"ECC ENABLE" IS ONLY IMPLEMENTED AT THE BEGINNING OF THE PROGRAM (I.E., CHANGING THE CF'S VALUE IN THE PROGRAM MONITOR WILL HAVE NO AFFECT UNTIL THE NEXT PASS OF THE PROGRAM).

7. ERROR LOG

THE ERROR LOG CONSISTS OF A LIST OF 4-WORD BLOCKS THAT CONTAIN THE FOLLOWING INFORMATION:

WORD #	CONTENTS
1	BOARD, BANK, MODULE, BIT NUMBER
2	RANGE OF PHYSICAL BLOCK ADDRESSES
3	TEST NAMES' BIT PATTERN
4	FAILURE COUNT

THE FAILURE COUNT IS THE TOTAL NUMBER OF TIMES AN ERROR HAS OCCURRED UNDER ALL TESTS RUN FOR THE CHIP WHOSE LOCATION IS SPECIFIED IN WORD 1.

LOGGING AN ERROR ENTAILS THE FOLLOWING ACTIONS:

A/ ERROR'S CHIP LOCATION IN LOG?

B/ IF IT IS IN THE LOG, UPDATE THE RANGE OF PHYSICAL ADDRESSES, THE TEST NAMES' BIT PATTERN, AND INCREMENT THE FAILURE COUNT.

C/ IF IT IS NOT IN THE LOG, CREATE A NEW NODE WITH ALL PERTINENT INFORMATION ENTERED FOR THAT ERROR, UPDATE LOG.END POINTER.

IF AN ATTEMPT TO CREATE ANOTHER NODE WOULD OVERFLOW THE LOG, THEN A WARNING MESSAGE IS ISSUED, AND THE PROGRAM MONITOR IS ENTERED.

FIGURE 2 IS AN EXAMPLE OF AN ERROR LOG PRINTOUT. SEE THE NOTES UNDER FIGURE 2 FOR MORE INFORMATION.

8. TEST PATTERNS

ENTER A BIT PATTERN TO DETERMINE WHAT TESTS ARE TO BE RUN. SEE TABLE 1 FOR TEST NAMES' BIT POSITIONS. A BIT SET WOULD MEAN THAT THAT TEST WILL BE RUN.

A NOTE ON THE OPTIONALLY AVAILABLE PATTERNS-

GALLOPING PATTERN IS A VERY SLOW BUT COMPREHENSIVE TEST PATTERN WHICH OFFERS STRONG TESTS FOR FAULTY ADDRESS DECODERS, ALL TYPES OF CELL INTERACTIVE COUPLING, SLOW ACCESS TIME, AND CELL FUNCTIONALITY.

GALLOPING READS TESTS MULTIPLE WRITING WITHIN ROWS AND REFRESH SENSITIVITY.

THE CHOSEN TESTS ARE THEN RUN UNDER THE DESIGNATED PARAMETERS OF THE MEMORY BOUNDED BY STADR AND ENDADR.

9. MULTIPROGRAMMING RELOCATION TEST

THIS TEST TREATS EACH BANK IN MEMORY AS IF IT WAS "THE ONLY BANK IN TOWN." (BAD PUN...) SERIOUSLY SPEAKING, IT PLACES THE PROGRAM IN THE SAME BANK THAT IS BEING TESTED AND DOES THIS FOR ALL THE BANKS IN MEMORY.

THIS TEST SHOULD BE RUN LAST AND ALONE (IT'S A DANGEROUS ANIMAL!). CHOOSING MULTIPROGRAMMING CANCELS MANUAL PROGRAM RELOCATION.

10. MANUAL PROGRAM RELOCATION (MPR)

IF ENABLED, THE USER ENTERS AN ADDRESS (MPR) TO WHERE THE PROGRAM WILL BE RELOCATED. THE RANGE MPR TO MPR+8K-1 WILL BE THE LOCATION OF THE PROGRAM DURING EXECUTION OF ALL THE CHOSEN TESTS.

MPR IS ONLY IMPLEMENTED AT THE BEGINNING OF THE PROGRAM (I.E., CHANGING THE CF'S VALUE IN THE PROGRAM MONITOR WILL HAVE NO AFFECT UNTIL THE NEXT PASS OF THE PROGRAM).

TO ENDADR; 2) MPR TO MPR+8K-1 MUST EXIST IN MEMORY; 3) IF RUNNING GALLOPING TEST PATTERNS, MPR TO MPR+8K-1 MUST NOT BE IN A BANK WHICH IS TO BE TESTED. REMEMBER THAT IS ENDADR OR STADR SPILLS OVER INTO ANOTHER BANK, THAT WHOLE BANK WILL BE TESTED. SEE TABLE 3 FOR THE SMALLEST AREA OF MEMORY THAT CAN BE TESTED FOR A PARTICULAR TEST. 4) DO NOT RUN MULTIPROGRAMMING WITH M.P.R.

FAILURE TO COMPLY WITH THESE WARNING WILL, IN MOST CASES, MAKE THE PROGRAM CRASH.

C) ERROR REPORTING

UPON HITTING AN ERROR, THE PROGRAM PRINTS OUT 1. 2. IS PRINTED OUT AT THE END OF EACH BOARD. 3. IS PRINTED OUT AT THE END OF THE PROGRAM.

IF THE SEI CF IS SET, 4. IS PRINTED ALONG WITH 1.

1. ERROR REPORT FORMAT

```
TEST NAME: _______________
BOARD NUMBER= _______
MODULE= __
BANK= __
BIT NUMBER(S)= _ _ _ _ _ _
```

BE APPENDED TO THE ERROR REPORT FORMAT.

2. NORMAL TERMINATION FORMAT

```
TESTING COMPLETED ...

BOARD NUMBER= _______.
ADDRESS START= ________
ADDRESS END= _________
RAM TYPE= ____ K
```

THE 'ADDRESS START' REFERRED TO IN THE NORMAL TERMINATION REPORT IS THE BOARD STARTING ADDRESS. THE 'ADDRESS END' IS THE BOARD ENDING ADDRESS.

3. END PROGRAM FORMAT

(MAKES SENSE, HUH?)

4. SUPPLEMENTAL ERROR INFORMATION

```
ADDRESS (PHYS./LOG)=.___   / _______
EXPECTED DATA= ________________
ACTUAL DATA= ________________
```

THE ADDRESS OF ERROR IS IN TWO PARTS, 'PHYS.' AND 'LOG.' . THE FIRST NUMBER IS THE PHYSICAL 1K BLOCK OF THE ADDRESS, AND THE SECOND THE WORD WITHIN THAT 1K BLOCK.

THE SUPPLEMENTAL ERROR INFORMATION IS PRINTED BETWEEN REPORT FORMAT.

D) PROGRAM MONITOR

IN BOTH MANUAL AND AUTO MODES, AT THE END OF EACH TEST, AT THE OCCURRENCE OF AN ERROR, AND AT THE END OF EACH PASS, THE TTY IS TESTED FOR INPUT. IF A VALID SWITCH PACK COMMAND WAS ENTERED, IT IS THEN EXECUTED. IF NOT, A CHECK FOR 'CNTRL-C' IS PERFORMED. IF 'CNTRL-C' WAS TYPED, A SMALL "PROGRAM MONITOR" IS EXECUTED. A "**?" PROMPT SIGNALS THE USER TO ENTER ONE OF THE VALID COMMANDS LISTED IN TABLE 4. ILLEGAL COMMANDS HAVE NO ILL EFFECTS.

ONCE IN THE PROGRAM MONITOR, SWITCH PACK COMMANDS ARE NOT ACCEPTED. FOR THIS REASON, IN ORDER TO GET A HARD COPY OF THE ERROR LOG PRINTOUT, IT IS NECESSARY TO ENABLE SWREG'S BIT 5 BEFORE ENTERING THE PROGRAM MONITOR (I.E., TYPE '5' BEFORE 'CNTRL-C' AND 'P' OR 'D').

ANY CHANGES MADE IN THE PATS, PSV, OR ECCE CF'S WILL NOT TAKE EFFECT UNTIL THE NEXT BOARD (FOR BOARDS 2 THRU LAST) OR THE NEXT BANK (FOR BOARD 1).

IV. DESIGN HIGHLIGHTS

A) ALL GALLOPING TEST PATTERNS WILL BE RUN USING A MICROCODE ROUTINE TO ACHIEVE THE GREATEST SPEED AND TO TEST THE MEMORY TO ITS FULLEST.

B) MEMORY SIZING WILL TAKE ADVANTAGE OF A MICROCODE ROUTINE THAT ACCESSES TWO MEMORY LOCATIONS FAST ENOUGH TO CAUSE MEMORY INTERFERENCE. USE OF THE REAL TIME CLOCK WILL COMPLETE THE MEMORY SIZING ALGORITHM.

C) EXISTING DLIB ROUTINES THAT ARE APPLICABLE ARE TO BE USED.

D) AUTOMATIC PROGRAM RELOCATION SCHEME

AS USUAL, THE PROGRAM WILL TEST THE MEMORY ON A BOARD BY BOARD BASIS. THE PROGRAM NEED ONLY BE RELOCATED WHEN TESTING THE FIRST BOARD. FOR BOARDS 2 THRU LAST, THE PROGRAM WILL RESIDE IN THE FIRST 4K OF MEMORY AND THE WHOLE BOARD UNDER TEST WILL BE USED FOR TESTING. HOWEVER, WHEN TESTING BOARD 1, THE LOCATION OF THE PROGRAM WILL VARY ACCORDING TO THE FOLLOWING SCHEMES:

1. 16K RAMS/128K BOARD: A)PROGRAM IN PHYSICAL BLOCKS (P.B.) 0-7, TEST P.B. 100-177. B) PROGRAM IN P.B. 100-107, TEST 0-77.

2. 4K RAMS/32K BOARD: A) PROGRAM IN P.B. 0-7, TEST 20-37. B) PROGRAM IN P.B. 20-27, TEST 0-17.

3. 16K RAMS/64K BOARD: A) PROGRAM IN P.B. 0-7, TEST 40-77. B) PROGRAM IN P.B. 40-47, TEST 0-37. C) PROGRAM IN P.B. 0-7, TEST 20-57. D) PROGRAM IN 20-27, TEST [10-17,60-77].

02
03 4. 4K N4MS/16K BOARDS: A) PROGRAM IN P.R. 0-7, TEST 8-15.
04 B) PROGRAM IN P.R. 8-15, TEST 0-7. C) PROGRAM IN P.R.
05 4-11, TEST (0-3,12-15). D) PROGRAM IN P.R. (0-3,12-15),
06 TEST 4-11.
07
08 E) INTERRUPTS WILL BE ENABLED ONLY FOR DETERMINING BOARD
09 CONFIGURATIONS AND DURING POWER SUPPLY VOLTAGE MARGINING WITHIN
10 THE PROGRAM EXEC. THEY WILL BE DISABLED THE REST OF THE PROGRAM.
11
12 F) IF THE MMPU1 DOES NOT EXIST, N4MDM WILL IMMEDIATELY RETURN
13 TO DTOS. IN THIS CASE, IT IS NECESSARY TO RUN N4MDU1 AND N4MDU2.
14
15
16
17 V. KNOWN ANOMALIES
18 ==========
19
20 A) ROW AND BIT COLUMN RIPPLING
21 ==========
22
23 1/ A BAD CHIP MAY LOAD DOWN A SIGNAL (E.G., RAS OR CAS)
24 SUFFICIENTLY TO CAUSE SOME OR ALL OF THE OTHER CHIPS IN THE SAME
25 BANK-MODULE (I.E.,ROW) OR BIT COLUMN TO GENERATE ERRORS. IN THE ERROR
26 LOG, THE GENUINE BAD CHIP CAN BE SPOTTED AS THE ONE WITH THE MOST
27 ERRORS (A FAILURE COUNT IN THE THOUSANDS), WHILE THE GOOD CHIPS
28 WOULD HAVE A FAILURE COUNT ONLY THE HUNDREDS. IF THIS OCCURS, REPLACE
29 THE CHIP WITH THE HIGHEST NUMBER OF REPORTED ERRORS AND RERUN THE
30 PROGRAM. IF ALL THE OTHER CHIPS WERE REALLY GOOD, THEN ALL THE
31 RELATED ERRORS SHOULD DISAPPEAR.
32
33 2/ YOU RUN THE PROGRAM WITH ERROR REPORTING (AS WOULD BE THE CASE
34 IN AUTO MODE). AN ERROR IS REPORTED, AND YOU REPLACE THE REPORTED
35 CHIP. IF, AFTER RERUNNING THE PROGRAM, YOU STILL GET AN ERROR IN THE
36 SAME ROW AND/OR THE SAME BIT COLUMN, THIS MAY BE THE DREADED 'RIPPLING'
37 EFFECT. INSTEAD OF COMMITING SUICIDE, RETURN THE OFFENDED CHIP (WITH
38 APPROPRIATE APOLOGIES) TO THE BOARD AND RERUN THE PROGRAM USING
39 ERROR LOGGING, FOLLOWING THE PROCEDURE OUTLINED IN '1/'.
40
41 3/ IF THE LOADING FROM THE GENUINE BAD CHIP IS SEVERE ENOUGH, IT
42 CAN PULL A CRITICAL SIGNAL DOWN OR UP AND CAUSE ALL THE "GOOD" CHIPS
43 IN THE SAME ROW AND/OR BIT COLUMN TO GENERATE AS MANY ERRORS AS THE
44 BAD CHIP ITSELF. IN THIS CASE, YOU'RE OUT OF LUCK, AND MUST RESORT
45 TO EMULATING A BIRD (IN OTHER WORDS, THE 'OL "HUNT AND PECK" METHOD).
46
47
48 B) IF YOU TYPE 'CNTRL-C' DURING AN ERROR REPORT WHEN ECM=3
49 (ENTER PROGRAM MONITOR UPON ERROR), WHEN IT COMES TIME TO 'EXIT
50 PROGRAM MONITOR', YOU WILL HAVE TO ENTER THE 'E' COMMAND TWICE-
51 ONCE FOR YOUR 'CNTRL-C' AND ANOTHER BECAUSE ECM=3.
52
53 C) IF YOU FIND AN ERROR IN THE SECOND BANK WHILE
54 RUNNING IN AUTO MODE, AND YOU WANT TO VERIFY IT USING ERROR
55 LOGGING, BE SURE TO IMPLEMENT STARTING AND ENDING ADDRESS
56 SO THAT TESTING IS RESTRICTED TO THE SECOND BANK ONLY (E.G.,
57 ON A 32K BOARD, STADR=20, ENDADR=37). IF THIS IS NOT DONE,

02
03 THE PROGRAM, WHEN FINISHED TESTING THE SECOND BANK, WOULD
04 RELOCATE THE PROGRAM IN THE SECOND BANK (IN ORDER TO TEST
05 THE FIRST BANK) AND WOULD THEREFORE CRASH/BOMB/SELF-DESTRUCT/
06 GO OUT TO LUNCH/FLY SOUTH.
07
08 D) THE N4MDM PROGRAM IS DESIGNED SPECIFICALLY TO RUN IN A
09 DTOS STREAM. IT IS NOT RECOMMENDED TO HALT THE PROGRAM AND
10 ATTEMPT TO CONTINUE PROCESSING AFTER EXECUTING SOME WORTHLESS
11 SOFT CONSOLE COMMANDS. IN THE MAJORITY OF THE CASES, IMPORTANT
12 DATA (E.G., BACKGROUND DATA IN THE MEMORY UNDER TEST) MAY BE
13 LOST, ALONG WITH THE MAPPING INFORMATION. IF IT IS NECESSARY
14 TO *RESTART* THE PROGRAM (I.E., "FROM THE BEGINNING"), THAT
15 IS O.K., AS LONG AS YOU DON'T ATTEMPT TO RETURN TO DTOS USING
16 THE PROGRAM MONITOR "A" COMMAND (IT WON'T WORK).
17
18
19 THE ABOVE REFERS TO THE PROGRAM MONITOR "H" COMMAND.
20 THE SAME WILL APPLY IF YOU PRESS THE MACHINE'S 'RESET' BUTTON.
21 IN THAT CASE, YOU'RE BETTER OFF RELOADING DTOS AND STARTING AGAIN.
22
23
24
25
26
27
28
29
30 ----------
31 ----------
32
33 TABLES AND FIGURES FOLLOW

10015 N4MDM

TABLE 1: TEST NAMES' BIT POSITIONS

BIT POSITION	TEST NAME	MODE RUN	ESTIMATED TEST TIME
15	MODIFIED DATA=ADDRESS	A	15 SEC
14	DATA=ADDRESS, UPPER	A	5 SEC
13	STUCK ADDRESS BIT	A	3 SEC
12	MARCHING 1/0	A	15 SEC
11	GALLOPING COLUMNS	A	1 MIN
10	GALLOPING ROWS	A	1 MIN
9	GALLOPING DIAGONAL	A	1 MIN
8	GALLOPING PATTERN	M	2 HOURS

NOTES: 1/ A=AUTO; M=MANUAL

2/ ALL ESTIMATED TEST TIMES ARE FOR A 256K WORD MEMORY, USING TWO 128K RAM BOARDS (WITH 16K RAMS).

3/ TEST TIMES FOR 4K RAM BOARDS WERE LESS THAN ONE MINUTE/32K BOARD WITHOUT GALLOPING PATTERN, AND ABOUT 2 MINUTES/32K BOARD WITH GALLOPING PATTERN.

4/ TEST TIMES LISTED ASSUME NO MEMORY ERRORS ARE DETECTED (I.E., IF THERE ARE ERRORS, AND ECM=4 (ERRORS BEING LOGGED, NOT PRINTED), THEN THE TESTS WILL TAKE LONGER TO COMPLETE).

10016 N4MDM

TABLE 2: CONTROL FLAGS' DEFAULT SETTINGS

DESCRIPTION OF CF	CF	PROGRAM OCTAL VALUE	DEFAULT
STARTING ADR. IN MEM.	STADR	0	BEG. OF MEMORY
ENDING ADR. IN MEMORY	ENDADR	TOP	TOP OF MEMORY
MODULES UNDER TEST	MUT	0	ALL
ERROR CONTROL MODE	ECM	0	PRINT ERROR, RETURN TO [illegible]TO
SUPPL. ERROR INFO.	SEI	0	NONE.
POWER SUPPLY VOLTAGES	PSV	0	NOMINAL ONLY
ECC ENABLE	ECCE	0	NOT AVAIL./NOT ENABLED
ERROR LOG PRINT	LP	0	NO LOG PRINT @ CONCL.
PATTERNS TO BE RUN	PATS	177	ALL EXCEPT GALPAT
MULTIPROGRAMMING	MULTI	0	NONE.
MANUAL PROG. RELOC.	MPR	0	DONE BY PROGRAM.

NOTES: 1/ CF=CONTROL FLAG

2/ THE DEFAULT PATTERNS ARE THE FIRST 7 PATTERNS LISTED LISTED IN TABLE 1.

3/ 'PROGRAM OCTAL VALUE' REFERS TO THE VALUE THE 'CF' WOULD CONTAIN TO IMPLEMENT THE DESCRIBED DEFAULT.

10017 V4M04

TABLE 3: SMALLEST AREA OF MEMORY THAT CAN BE TESTED,
=======   LISTED BY TEST NAME

TEST NAME	SMALLEST AREA
----------	--------------
MODIFIED DATA&ADDRESS	1&K BLOCK
DATA& ADDRESS UPPER	STA0N TO FNDA0N (1K BLOCK)
STUCK ADDRESS BIT	16K BLOCK WITHIN 1 MODULE
MARCHING 1/0	16K BLOCK WITHIN 1 MODULE
ALL GALLOPING TEST PATTERNS	1 MODULE WITHIN 1 BANK

10018 V4M0M

TABLE 4: PROGRAM MONITOR COMMANDS
=======

COMMAND	(STANDS FOR)	ACTION TAKEN
=======	============	============
A	ABORT	RETURN TO DTOS
C	CLEAR	CLEAR ERROR LOG
D	DUMP	PRINT & CLEAR ERROR LOG
E	EXIT	RETURN TO MAIN PROGRAM
F	FLAGS	PRINT CONTROL FLAGS
H	HALT	PROGRAM HALTED
P	PRINT	PRINT ERROR LOG
R	RESET	PRINT CF'S; INPUT NEW VALUES
T	TERMINATE	TERMINATE THE CURRENT TEST, AND RETURN TO THE PROGRAM EXEC.

NOTES: 1/ AN ILLEGAL COMMAND WILL CAUSE A MESSAGE TO BE
-----   PRINTED, LISTING THE VALID COMMANDS THAT MAY
           BE ENTERED.

        2/ FOR OBVIOUS REASONS, DO NOT ATTEMPT TO "ABORT" TO
           DTOS IF YOU HAVE HALTED AND SUBSEQUENTLY RESTARTED
           THE PROGRAM (WHICH IS NOT RECOMMENDED IN ANY CASE).
           IT WON'T WORK.

10019 N&MDM

FIGURE 1: FAULT TYPES VS. MEMORY TEST PATTERNS

TEST NAME	MAJOR FAULT TYPES TESTED
MODIFIED D=A	1,2,8
D=A UPPER	1,2,8
STUCK ADR. BIT	1,2
MARCHING 1/0	2,8
GALLOPING COL.	4,5
GALLOPING ROWS	3,5,7
GALLOPING DIAG.	3,4,6,7
GALLOP'G PATTERN	1,2,3,4,6,7,8

KEY:

1: FAULTY ADDRESS DECODING- WRONG CELL OR CHIP ACCESSED
2: FAULTY ADDRESS DECODING- CELL INACCESSIBLE
3: INTERACTIVE COUPLING WITHIN ROWS
4: INTERACTIVE COUPLING WITHIN COLUMNS
5: REFRESH SENSITIVITY
6: SLOW ACCESS TIME
7: SENSE AMPLIFIER RECOVERY
8: CELL FUNCTIONALITY

10020 J&40M

FIGURE 2: ERROR LOG OUTPUT

____CHIP LOCATION____ BOARD	BANK	MOD	BIT	ADR. RANGE LOW	HIGH	TEST NAMES	FAILURE COUNT
1	1	C	9	102	137	000140	360
1	1	C	12	102	137	000140	360
1	1	D	0	105	105	000140	32157
1	0	A	6	75	77	000202	3
2	0	A	9	200	377	000040	64
2	0	H	10	200	277	000040	64
2	0	B	15	200	377	000040	64

NOTES:

1/ THE NUMBERS LISTED UNDER 'ADR. RANGE' ARE PHYSICAL 1K BLOCK NUMBERS. THEY ARE THE RANGE IN WHICH ERRORS OCCURRED ON THE SPECIFIED CHIP. (E.G., 102-137 MEANS THAT ERRORS OCCURRED FROM P.A. 102 TO PA. 137; 105-105 WOULD MEAN THAT ERRORS OCCURRED ONLY WITHIN P.A. 105.)

2/ 'TEST NAME(S)' IS A BIT PATTERN, A BIT SET MEANS THAT AN ERROR HAS OCCURRED UNDER THE CORRESPONDING TEST LISTED IN TABLE 1 FOR THE SPECIFIED CHIP LOCATION. (E.G., 41= BITS 10,15 => GALLOPING ROWS AND MODIFIED DATA=ADDRESS, IF RUNNING THE MAPPED VERSION). BIT 0 SET MEANS THAT ECC WAS ENABLED DURING TESTING.

3/ THE TEST NAMES LISTED INDICATE UNDER WHAT TESTS AN ERROR HAS OCCURRED IN THE WORD.

4/ 'FAILURE COUNT' IS THE NUMBER OF TIMES AN ERROR HAS OCCURRED AT THE SPECIFIED CHIP LOCATION.
IF THE FAILURE COUNT=177777, THEN THERE WAS AN OVERFLOW IN THE FAILURE COUNT DATA WORD (I.E., THE NUMBER OF ERRORS FOR THAT CHIP WAS >=177777).

5/ ALL NUMBERS ARE IN OCTAL.

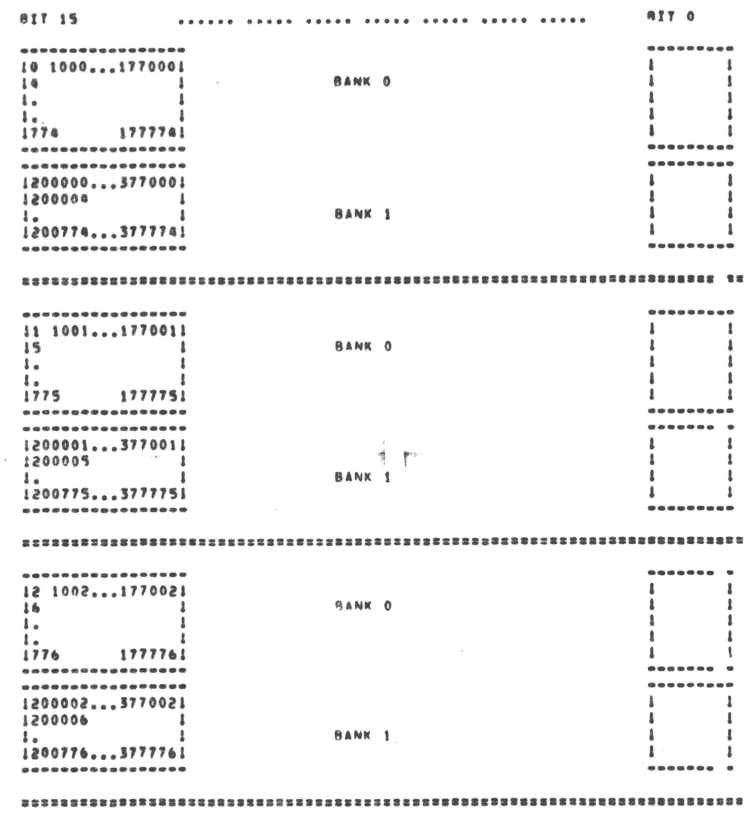
6/ "ADR. RANGE/LOW, HIGH" IS NOT INCLUDED IN THE UNMAPPED VERSIONS.

7/ TO GET A HARD COPY OF THE ERROR LOG PRINTOUT, IT IS NECESSARY TO SET SWREG'S BIT 5 BEFORE ENTERING THE PROGRAM MONITOR (UNLESS, OF COURSE, THE CONSOLE DEVICE IS A PRINTER OR TELETYPE). IF THE LOG IS TO BE PRINTED OUT ONLY AT THE CONCLUSION OF THE PROGRAM, SET BIT 5 ANYTIME DURING THE EXECUTION OF THE PROGRAM.

8/ A 'GALLOPING PATTERN' FAILURE COUNT OF 1 SHOULD BE INTERPRETED AS A HARD ERROR. ("1" IS THE MAXIMUM FAILURE COUNT THAT GALPAT WILL GENERATE).

10023 NAMDM

FIGURE 41 MEMORY TOPOLOGY, 16K RAMS



10024 NAMDM

EACH BLOCK REPRESENTS A CHIP. THE NUMBERS WITHIN THE BLOCK DENOTE PHYSICAL ADDRESSES.

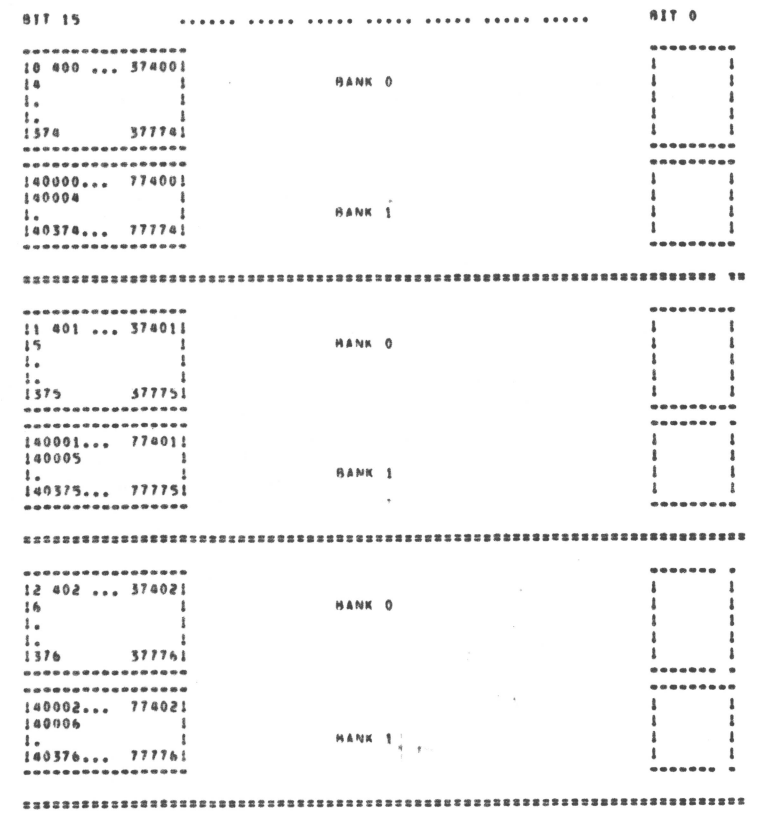
THIS IS THE VIEW IF THE FRONT END (I.E., THE ONE WITH THE PINS) IS TOWARDS YOU. THE BANKS ALTERNATE; THE MODULES RUN BACK TO FRONT (A-D); THE BIT NUMBERS RUN RIGHT TO LEFT. IF THERE ARE ANY ECCC CHIPS, THEY WOULD BE APPENDED TO THE LEFT.

A 128K BOARD WOULD HAVE ONLY BANK 0. A 256K BOARD WOULD HAVE BOTH BANK 0 AND BANK 1.

.ENDC
.EJEC

10021 NAMDM

FIGURE 3: MEMORY TOPOLOGY, 4K RAMS



10022 NAMDM

EACH BLOCK REPRESENTS A CHIP. THE NUMBERS WITHIN THE BLOCK DENOTE PHYSICAL ADDRESSES.

THIS IS THE VIEW IF THE FRONT END (I.E., THE ONE WITH THE PINS) IS TOWARDS YOU. THE BANKS ALTERNATE; THE MODULES RUN BACK TO FRONT (A-D); THE BIT NUMBERS RUN RIGHT TO LEFT. IF THERE ARE ANY ENCC CHIPS, THEY WOULD BE APPENDED TO THE LEFT.

A 16K BOARD WOULD HAVE ONLY BANK 0, A 32K BOARD WOULD HAVE BOTH BANK 0 AND BANK 1.

```
 0001 .MAIN    MACRO REV 06.30          00:03:35 12/27/78
01
02
03
04
05
06              ;*****************************************************************
07              ;
08              ; NAME: N4EXER.SR                          PART NUMBER: 094-001539
09              ;
10              ;
11              ; DESCRIPTION: EXERCISER FOR NOVA 4 TEST.
12              ;
13              ;
14              ; REVISION HISTORY:
15              ;
16              ;       REV.            DATE
17              ;
18              ;       00   /          12/22/78
19              ;
20              ;
21              ; COPYRIGHT (C) DATA GENERAL CORPORATION 1978.
22              ; ALL RIGHTS RESERVED.
23              ; LICENSED MATERIAL - PROPERTY OF DATA GENERAL CORPORATION.
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      000000          .TXTM   0
```

```
10002 .MAIN
01
02
03          102032 .DALC SGE = AOCZ# 0,0,SZC
04          102432 .DALC SGT = SU6Z# 0,0,SZC
05
06
07
08                  ;NOVA3 INSTRUCTION SET
09
10          061401 .DIAC PSH = 61401
11          061601 .DIAC POP = 61601
12          061001 .DIAC MTSP = 61001
13          060001 .DIAC MTFP = 60001
14          061201 .DIAC MFSP = 61201
15          060201 .DIAC MFFP = 60201
16          062401 .DUSR SAVE = 62401
17          062601 .DUSR RTRN = 62601
18          100010 .DUSR .SCL = 100010
19          000002 .DUSR MMU  = 2           ;MAP DEF.
20          000004 .DUSR PRTY = 4           ;PARITY DEF.
21          060104 .DUSR CODP = NIOS PRTY
22          060204 .DUSR CLRP = NIOC PRTY
23          073301 .DUSR MUL= 73301
24          077201 .DUSR MULS= 77201
25          073101 .DUSR DIV= 73101
26          077001 .DUSR DIVS= 77001
```

NOVA 4 EXERCISER

```
 0001  PNCST    AOS ASSEMBLER REV 02.02          15:27:44 01/02/79
01
02
03
04
05
06
07              ;*****************************************************************
08              ;
09              ; NAME: N4LGCTST.TX                          PART NUMBER: 097-001137
10              ;
11              ;
12              ; DESCRIPTION: NOVA 4 LOGIC TEST
13              ;
14              ;
15              ; REVISION HISTORY:
16              ;
17              ;       REV.            DATE
18              ;
19              ;       00              12/22/78
20              ;
21              ;
22              ; COPYRIGHT (C) DATA GENERAL CORPORATION 1978.
23              ; ALL RIGHTS RESERVED.
24              ; LICENSED MATERIAL - PROPERTY OF DATA GENERAL CORPORATION.
25              ;*****************************************************************
```

N4 Logic

WARNING
N4LGCTST DOES NOT SUPPORT THE DIAGNOSTIC SWITCH PACKAGE. TYPING ON THE TELETYPE MAY CAUSE PROGRAM ERRORS

```
10002  PNCST
01
                        .TITL N4LGCTST
05              ;               NOVA 4 LOGIC TEST
04              ;1.     ABSTRACT
05              ;               THE NOVA 4 LOGIC TEST IS A MAINTENANCE PROGRAM
06              ;               DESIGNED TO TEST THE NOVA 4 CENTRAL PROCESSING
07              ;               UNIT.  IT IS A GATE BY GATE TEST OF THE LOGIC
08              ;               USED TO IMPLEMENT THE NOVA 4 INSTRUCTION SET.
09              ;               ALSO INCLUDED IS A MINIMUM LEVEL TEST OF THE
10              ;               CPU I/O INSTRUCTIONS,TELETYPE I/O, AND
11              ;               PROGRAM INTERRUPT.
12              ;
13              ;2.     MACHINE REQUIREMENTS
14              ;2.1            NOVA 4 PROCESSOR
15              ;2.2            8K OF READ/WRITE MEMORY
16              ;2.3            BASIC I/O TELETYPE INTERFACE
17              ;
18              ;
19              ;4.     OPERATING PROCEDURE
20              ;4.1            VERIFY THAT THE NOVA 4 WILL PERFORM ALL
21              ;               CONSOLE FUNCTIONS. I.E. EXAMINE/EXAMINE NEXT
22              ;               DEPOSIT/DEPOSIT NEXT AC'S EXAMINE/DEPOSIT
23              ;4.2            LOAD THE PROGRAM VIA THE BINARY LOADER.
24              ;4.3            SET THE SWITCHES EQUAL TO 200
25              ;4.4            PRESS START
26              ;4.5            MACHINE SHOULD HALT M/A=201. PRESS CONTINUE
27              ;4.6            PROCESSOR SHOULD CONTINUE TO RUN WITHOUT HALTING
28              ;               TELETYPE SHOULD STUTTER FOR 60 CHARACTERS
29              ;               THE TYPEOUT "PASS" SHOULD OCCUR AND THE TEST
30              ;               SHOULD CONTINUE TO LOOP WITH THE TELETYPE RUNNING
31              ;               AT A SLOWER RATE.
32              ; 4.7          TO RESTART AFTER FIRST PASS, START AT LOC 170
33
34
35              ;5.     ERROR DESCRIPTION
36              ;5.1            DETECTED ERRORS WILL CAUSE THE PROGRAM TO DO A
37              ;               PROCESSOR HALT.
38              ;5.2            RECORD THE STATE OF THE PROCESSOR AND REGISTERS
39              ;               AT THE TIME OF THE HALT.  CONSULT THE LISTING
40              ;               AT THE ADDRESS OF THE ERROR HALT FOR PROB-
41              ;               ABLE CAUSES OF THE FAILURE. CONSTRUCT A LOOP
42              ;               THAT WILL REPEAT THE FAILURE AND SCOPE AS REQUIRED.
43
44              ;6.     PROGRAM DESCRIPTION
45              ;               THIS PROGRAM IS A COLLECTION OF SMALL TESTS,
46              ;               EACH TEST IN SEQUENCE BASED ON PREVIOUS TESTS
47              ;               WORKING AND DESIGNED TO TEST AS SMALL AN ADDI-
48              ;               TIONAL PIECE OF THE LOGIC AS POSSIBLE.
49              ;
50              ;
51              ; 7.     CAT/KITTEN OPERATION
52              ;
53              ;               IF THE PROGRAM WAS LOADED FROM DTOS WITH CAT
54              ;               OR KITTEN THE PROGRAM WILL RUN IT IN
55              ;               THE BACKGROUND AFTER ONE PASS OF USING THE TTY
56              ;               INTERRUPTS. THE PROGRAM WILL RUN MUCH
57              ;               SLOWER ALLOWING THE CAT/KITTEN AMPLE TIME TO
58              ;               COMPLETE A PASS.
59              ;
60              ;
```