



DATA GENERAL  
CORPORATION

Southboro,  
Massachusetts 01772  
(617) 485-9100

PROGRAM

Memory Checkerboard III

TAPES

Binary 095-000031-02

ABSTRACT

Memory Checkerboard III is a maintenance program designed to produce worst case noise conditions on the sense/inhibit wires. The program should be run to insure proper operation of sense amps, inhibit drivers, and memory currents.



01  
02  
03  
04  
05

07  
08  
09  
10  
11  
12  
13  
14  
15  
16  
17  
18  
19  
20  
21  
22  
23  
24  
25  
26

```
*****  
/ NAME: CHECKIII.SR PART NUMBER: 094-000031  
/ DESCRIPTION: CHECKERBOARD III  
/ REVISION HISTORY:  
/ REV. DATE  
/ 00 05/07/70  
/ 01 06/28/71  
/ 02 01/25/74  
/ COPYRIGHT (C) DATA GENERAL CORPORATION, 1970, 1971, 1974  
/ ALL RIGHTS RESERVED.  
*****
```

01  
02  
03  
04  
05  
06  
07  
08  
09  
10  
11  
12  
13  
14  
15  
16  
17  
18  
19  
20  
21  
22  
23  
24  
25  
26  
27  
28  
29  
30  
31  
32  
33  
34  
35  
36  
37  
38  
39

## CHECKERBOARD III

## 11. ABSTRACT

CHECKERBOARD III IS A MAINTENCE PROGRAM DESIGNED TO PRODUCE WORST CASE NOISE CONDITIONS ON THE SENSE/INHIBIT WIRES. THE PROGRAM SHOULD BE RUN TO INSURE PROPER OPERATION OF SENSE AMPS, INHIBIT DRIVERS, AND MEMORY CURRENTS.

## 12. MACHINE REQUIREMENTS

12.1 NOVA/SUPERNOVA PROCESSOR  
12.2 4K READ/WRITE MEMORY

## 13. SWITCH SETTINGS

13.1 STARTING ADDRESS =000002  
13.2 SWITCH 0(1) =1024 READ/WRITE DISTURB  
13.3 SWITCH 15(1) =INHIBIT HALT ON ERROR

## 14. OPERATING PROCEEDURE

14.1 LOAD THE PROGRAM VIA THE BINARY LOADER  
14.2 SET SWITCHES TO 000002  
14.3 PRESS START  
14.3.1 THE PROGRAM WILL PRINT THE HIGHEST LOCATION THE PATTERN IS TO USE.  
14.4 IF THE FAILURES ARE MARGINAL, SETTING SWITCH 0 MAY AID IN INDUCING A FAILURE TO OCCURE.  
14.5 WHEN SCOPING OR ADJUSTING CURRENTS, SETTING SWITCH 15 WILL INHIBIT THE ERROR HALT. THE BELL WILL STILL BE RUNG.  
14.6 PROGRAM MODIFICATIONS  
14.6.1 C(3)=ADR THE STARTING PATTERN ADD  
14.6.2 C(5)=INHIBIT INHIBIT THE CHECKERBOARD PATTERN ON CLEARED BITS.

A 0003 .MAIN

```
01
02      15.      PROGRAM OUTPUT/ERROR DISCRIPTION
03      15.1     AT EACH OCCURANCE OF ERROR, IF THE TELETYPE IS N
04      )       BUSY THE BELL WILL BE RUNG. IF SWITCH (15) IS ZE
05      )       THE PROGRAM WILL HALT AT LOCATION "ER".
06      )       15.2     WHEN A ERROR HALT OCCURES:
07      )       )       C(1)=THE ERROR WORD
08      )       )       C(2)=THE ERROR ADDRESS
09      )       )       C(3)=ADDRESS OF PROGRAM
10      )       15.3     SET SWITCH (15) IF SCOPING,PRESS CONTINUE.
11      )       15.4     SYNC PULSES
12      )       )       A "P" PULSE (A74) IN STORE CYCLE.
13      )       )       A "S" PULSE (A52) CHECK ONES PATTERN WOR
14      )       )       A "C" PULSE (A50) CHECK ZEROS PATTERN WO
15
16      16.      PROGRAM DISCRIPTION
17      16.1     STORE THE CHECKERBOARD PATTERN
18      16.2     IF SWITCH 0(1) DISTURB THE CONTENTS OF MEMORY BY
19      )       REFFERANCING LOCATIONS 0101,0202,0303,ETC. 512
20      )       TIMES. THIS PRODUCES 1024 READ/WRITE DISTURBS.
21      )       16.3     CHECK THE PATTERN WORD
22      )       16.4     COMPLEMENT AND CHECK THE WORD
23      )       16.5     RESTORE THE WORD
24      )       16.6     WHEN THE END OF THE PATTERN IS REACHED THE
25      )       PROGRAM COMPLEMENTS THE PATTERN WORD AND RE-
26      )       TURNS TO STEP 6.1 .
27
28      17.      LIMITATIONS
29      )       NONE
30
```

A 0004 .MAIN

01  
02 000002 .LOC 2  
03 000002 000164 JMP MSIZ

04  
05 000003 000620 ADR: 620  
06 000004 007577 FINAL: 7577  
07 000005 177777 INH: -1  
08 000006 000000 PATT: 0  
09 000007 000000 ERET: 0

!PATTERN STARTING ADDRESS  
!PATTERN FINAL ADDRESS  
!MASK FOR INHIBITED BITS  
!PATTERN WORD

10  
11 000040 .LOC 40

12  
13 000400 000017 C17: 17  
14 000401 000400 C400: 400  
15 000402 000077 C77: 77  
16 000403 007777 C7777: 7777  
17 000404 000207 C207: 207  
18 000405 000101 C101: 101  
19 000406 060200 CNIOC: NIOC 0  
20 000407 060100 CNIOS: NIOS 0  
21 000500 070000 C070000: 070000  
22 000501 000000 MODUAL: 0  
23 000502 000000 EDIST: 0  
24 000503 001000 K1000: 1000  
25 000504 177577 M201: -201  
26 000505 000224 CMA: MMA2  
27 000506 177625 PSIZE: BEGIN-CEND  
28 000507 000000 PLOC: 0  
29 000600 000000 BPROG: 0  
30 000601 000000 EPROG: 0  
31 000602 012345 RANDOM: 12345  
32 000603 177760 M20: -20  
33 000604 000020 C20: 20  
34 000605 000432 CBEG: BEGIN  
35 000606 000000 RETURN: 0

36  
37  
38 000607 004126 SEND: JSR .RAND !GET A RNADOM #

39 000700 000062 RANDOM  
40 000701 024004 LDA 1,FINAL  
41 000702 030056 LDA 2,PSIZE  
42 000703 147000 ADD 2,1  
43 000704 101220 MOVZR 0,0  
44 000705 122422 SUBZ 1,0,SZC  
45 000706 000075 JMP .-1  
46 000707 123000 ADD 1,0  
47 001000 024063 LDA 1,M20  
48 001001 123400 AND 1,0  
49 001002 024003 LDA 1,ADR  
50 001003 122433 SUBZ# 1,0,SNC  
51 001004 121000 MOV 1,0

!AC0=# MODULO C(FINAL).

!IF # TO SMALL USE  
!C(ADR) FOR STARTER.

52  
53 001005 040057 STA 0,PLOC  
54 001006 034063 LDA 3,M20  
55 001007 117000 ADD 0,3  
56 001100 054060 STA 3,BPROG  
57 001101 142400 SUB 2,0  
58 001102 040061 STA 0,EPROG

A 0005 .MAIN

```
01
02 00113 145000 MOVEI  MOV 2,1
03 00114 030000 LDA 2,CBEG
04 00115 021000 LDA 0,0,2 I MOVE A COPY OF
05 00116 041420 STA 0,20,3 ICHECKERBOARD TO
06 00117 175400 INC 3,3 ISELECTED SPOT.
07 00120 151400 INC 2,2
08 00121 125404 INC 1,1,8ZR ITEST FOR LAST REG
09 00122 000115 JMP MOVE+2 ITO BE MOVED.
10 00123 034057 LDA 3,PLOC
11 00124 005401 JSR 1,3 IEXIT TO PROG.
12 00125 000212 JMP MMA
13
14 00126 054100 .RAND: STA 3,.UD03 IGENERATE A RANDOM #
15 00127 010100 ISZ .UD03
16 00130 044100 STA 1,.UD01
17 00131 050107 STA 2,.UD02
18 00132 031400 LDA 2,0,3
19 00133 021000 LDA 0,0,2
20 00134 004143 JSR .UD50
21 00135 034102 LDA 3,.UD20
22 00136 103000 ADD 3,0
23 00137 041000 STA 0,0,2
24 00140 024100 LDA 1,.UD01
25 00141 030107 LDA 2,.UD02
26 00142 002100 JMP 0,.UD03 IRETURN
27
28 00143 024103 .UD50: LDA 1,.UD21
29 00144 044101 STA 1,.UD10
30 00145 105120 MOVZL 0,1
31 00146 125120 MOVZL 1,1
32 00147 014101 DSZ .UD10
33 00150 000140 JMP .-2
34 00151 107000 ADD 0,1
35 00152 125120 MOVZL 1,1
36 00153 125120 MOVZL 1,1
37 00154 123000 ADD 1,0
38 00155 001400 JMP 0,3
39
40 00156 000000 .UD01: 0
41 00157 000000 .UD02: 0
42 00160 000000 .UD03: 0
43 00161 000000 .UD10: 0
44 00162 033031 .UD20: 33031
45
46 00163 000010 .UD21: 10
47
```

A 0000 .MAIN

```

01
02 00164 020053 MSIZ1: LDA 0,K1000 I SIZE THE MEMORY
03 00165 115000 MOV 0,3
04 00166 031400 MSIZ1: LDA 2,0,3 I SAVE C(MEM)
05 00167 055400 STA 3,0,3
06 00170 025400 LDA 1,0,3
07 00171 051400 STA 2,0,3 I RESTORE MEMORY
08
09 00172 125014 MOV# 1,1,SZR
10 00173 124015 COM# 1,1,SNR
11 00174 000202 JMP MSIZ2 I END OF MEMORY
12 00175 136414 SUB# 1,3,SZR I AC1=BAD, AC3=GOOD
13 00176 063077 HALT I MEMORY FAILED.
14 00177 117000 ADD 0,3
15 00200 175113 MOVL# 3,3,SNC I INCREMENT MEMORY ADDRESS.
16 00201 000166 JMP MSIZ1 I TEST FOR 32K.
17
18 00202 020054 MSIZ2: LDA 0,M201
19 00203 163000 ADD 3,0
20 00204 040004 STA 0,FINAL
21 00205 004372 JSR CRLF
22 00206 004247 JSR MESS
23 00207 000234 MESIZE
24 00210 024004 LDA 1,FINAL
25 00211 004266 JSR POCT
26
27 00212 063611 MMA1: SKPDN TTO
28 00213 000212 JMP .-1 I WAIT FOR TTO DONE.
29 00214 020055 LDA 0,CMA
30 00215 040001 STA 0,1 I SET INTERRUPT RETURN.
31 00216 152520 SUBZL 2,2 I THIS PROG TEST FOR INTERRUPT
32 00217 102000 MMA1: ADC 0,0 I ABILITY TO CLEAR MA.
33 00220 040000 STA 0,0
34 00221 025000 LDA 1,0,2
35 00222 060177 NIOS CPU I ENABLE INTERRUPT
36 00223 005000 JSR 0,2 I SET BIT INTO MA
37 00224 045000 MMA2: STA 1,0,2
38 00225 020000 LDA 0,0
39 00226 112414 SUB# 0,2,SZR I AC0=PC STORED
40 00227 063077 HALT I AC2=CORRECT. MEMORY FAILED!
41 00230 151120 MOVZL 2,2
42 00231 151113 MOVL# 2,2,SNC
43 00232 000217 JMP MMA1
44 00233 000067 JMP SEND
45
46 MESIZE: .TXTE I
47 00234 040714 LAST LOCATION TESTED I
00235 152123
00236 146240
00237 141717
00240 152101
00241 147711
00242 120116
00243 142724
00244 152123
00245 042305
00246 000240

```



A 0007 .MAIN

```
01
02      )ITTO NON INTERRUPT PACKAGE
03      )"MESS" PRINTS ASCII MESSAGES AS SPECIFIED BY ASSEMBLR
04      )"CHAR" PRINTS ASCII CHARACTER, C(0)R,C(0)L MUST BE 0
05      )WILL RETURN +2 IF C(0)R=0,CORRECTS THE PARITY,33 SIMULATE
06      )"TYPE" PRINTS C(0)R. MUST HAVE PROPER PARITY. RETURN IS
07      )TO CALL+1.REPLACE THIS ROUTINE WITH INTERRUPT TYPE IF DESIRED.
08      )"CRLF" PRINTS A CARRIAGE RETURN
09      )"POCT" PRINTS C(1) IN OCTAL FOLLEOWED BY A TAB
10      )"PDEC" PRINTS C(1) IN DECIMAL,LEADONG ZEROS SUPPRESSED,
11      )FOLLOWED BY A TAB.
12
13 00247 054546 MESS:   STA 3,MESSR      )PRINT A TEXT MESSAGE
14 00250 010545      ISZ MESSR
15 00251 031400      LDA 2,0,3      )C(2) POINTS TO MESSAGE
16 00252 024542      LDA 1,C377     )A 8 BIT MASK
17 00253 021000      LDA 0,0,2      )C(2)=DATA WORD
18 00254 125112      MOVL# 1,1,SZC
19 00255 123701      ANDS 1,0,SKP
20 00256 123401      AND 1,0,SKP     )C(0)=DATA CHARACTER RIGHT
21 00257 151400      INC 2,2      )INC TO NEXT WORD
22 00260 124000      COM 1,1      )FLIP MASK
23 00261 004343      JSR CHAR      )PRINT
24 00262 000253      JMP MESS+4     )ANOTHER
25 00263 002532      JMP 0MESSR    )LAST
26
27 00264 020525 ZOCT:   LDA 0,CH240
28 00265 101001      MOV 0,0,SKP
29
30 00266 020525 POCT:   LDA 0,C60
31 00267 030322      LDA 2,OCTAB    )PRINT C(1) IN OCTAL
32 00270 000273      JMP 0+3
33 00271 030332 PDEC:   LDA 2,DECTB     )PRINT C(1) IN DECIMAL
34 00272 020517      LDA 0,CH240    )SUPPRESS LEADING ZEROS
35 00273 054342      STA 3,RADRET  )BOTH ENTRYS PRINT NUMBER
36 00274 040341      STA 0,ZSUPP   )THEN TAB TO NEXT POSITION
37 00275 050276      STA 2,0+1
38 00276 000000 DECOCT: 0      )A"LDA 2,TABLE" INSTRUCTION
39 00277 010276      ISZ 0-1
40 00300 034342      LDA 3,RADRET  )SETUP "TAB" AT END
41 00301 020503      LDA 0,CHTAB
42 00302 151005      MOV 2,2,SNR   )IF TABLE ENTRY=0
43 00303 000343      JMP CHAR      )EXIT WITH TAB
44 00304 034341      LDA 3,ZSUPP   )ZEROS SUPPRESS STUF
45 00305 102400      SUB 0,0
46 00306 146512 DECOT:  SUBL# 2,1,SZC
47 00307 000314      JMP DECP
48 00310 146400      SUB 2,1      )FORM THE DIGIT
49 00311 034502      LDA 3,C60
50 00312 101400      INC 0,0
51 00313 000306      JMP DECOT
52 00314 151235 DECP:   MOVZR# 2,2,SNR
53 00315 034476      LDA 3,C60
54 00316 054341      STA 3,ZSUPP   )C(0)=DIGIT
55 00317 163000      ADD 3,0      )MAKE ASCII
56 00320 004343      JSR CHAR      )PRINT
57 00321 000276      JMP DECOCT    )GET NEXT DIGIT
58
```

```

A 0008 .MAIN
01
02
03 00322 030323 OCTAB: LDA 2, .+1
04 00323 100000          100000
05 00324 010000          10000
06 00325 001000          1000
07 00326 000100          100
08 00327 000010          10
09 00330 000001          1
10 00331 000000          0
11
12 00332 030333 DECTB: LDA 2, .+1
13          000012 .RDX 10
14 00333 023420          10000
15 00334 001750          1000
16 00335 000144          100
17 00336 000012          10
18 00337 000001          1
19 00340 000000          0
20          000010 .RDX 8
21
22 00341 000000 ZSUPP: 0
23 00342 000000 RADRET: 0
24
25 00343 054442 CHAR: STA 3, CHRET ;PRINT C(0) RIGHT
26 00344 101325          MOVZS 0, 0, SNR ;RETURN +2 IF NULL
27 00345 001401          JMP 1, 3
28 00346 040440          STA 0, CHSAV
29 00347 176000          ADC 3, 3 ;COMPUTE THE PARITY
30 00350 117000          ADD 0, 3
31 00351 163404          AND 3, 0, SZR
32 00352 000347          JMP .-3
33 00353 176660          SUBCR 3, 3 ;COMBIND PARITY WITH CHAR
34 00354 020432          LDA 0, CHSAV
35 00355 163300          ADDS 3, 0
36
37 00356 034426 CHAR1: LDA 3, CHTAB ;IS THIS A TAB
38 00357 116405          SUB 0, 3, SNR
39 00360 000363          JMP .+3 ;YES
40 00361 004435          JSR TYPE ;NO PRINT IT
41 00362 002423          JMP @CHRET ;EXIT
42
43 00363 020424          LDA 0, CHORZ ;SIMULATE A TAB
44 00364 034424          LDA 3, CHAR7 ;VIA 1 TO 8 SPACES
45 00365 117405          AND 0, 3, SNR
46 00366 002417          JMP @CHRET
47 00367 020422          LDA 0, CH240
48 00370 004426          JSR TYPE
49 00371 000363          JMP .-6
50

```

A 0009 .MAIN

```
01
02
03
04 00372 054420 CRLF: STA 3,CRLF      ;SAVE RETURN
05 00373 020410 LDA 0,C215
06 00374 004343 JSR CHAR      ;PRINT CARRIAGE AND LF
07 00375 020405 LDA 0,C212
08 00376 004343 JSR CHAR
09 00377 102400 SUB 0,0
10 00400 040407 STA 0,CHORZ   ;CLEAR HORZ POSISTION
11 00401 002411 JMP @CRLF     ;EXIT
12
13 00402 000212 C212: 212
14 00403 000215 C215: 215
15 00404 000011 CHTAB: 11
16 00405 000000 CHRET: 0
17 00406 000000 CHSAV: 0
18 00407 000000 CHORZ: 0
19 00410 000007 CHAR7: 7
20 00411 000240 CH240: 240
21 00412 000000 CRLF: 0
22 00413 000060 C60: 60
23
24 00414 000377 C377: 377
25 00415 000000 MESSR: 0
26 00416 054412 TYPE: STA 3,TYPRET   ;TYPE THE C(0)R IF
27 00417 010770 ISZ CHORZ
28 00420 074477 READS 3   ;SWITCH 1(0).
29 00421 175100 MOVL 3,3
30 00422 175102 MOVL 3,3,SZC
31 00423 002405 JMP @TYPRET ;INHIBIT TYPE EXIT.
32 00424 063511 SKPBZ TTD
33 00425 000777 JMP .-1
34 00426 061111 DOAS 0,TTD
35 00427 002401 JMP @TYPRET
36 00430 000000 TYPRET: 0
37
38
```

A 0010 .MAIN

```
01
02
03 00431 063077 HALT ;OPERATOR ERROR FIX C(ADR)
04 00432 004401 BEGIN: JSR .+1
05 00433 054066 STA 3,RETURN
06 00434 034003 LDA 3,ADR
07 00435 030050 LDA 2,C070000
08 00436 020004 LDA 0,FINAL
09 00437 143400 AND 2,0
10 00440 040052 STA 0,EDIST
11 00441 173400 AND 3,2
12 00442 050051 STA 2,MODUAL ;THE MEMORY MODUAL
13
14 00443 030003 IPAT: LDA 2,ADR
15 00444 024041 LDA 1,C400
16 00445 020006 LDA 0,PATT ;PRESET PATTERN
17 00446 147404 AND 2,1,SZR
18 00447 100000 IPAT1: COM 0,0
19 00450 024005 LDA 1,INH
20 00451 123400 AND 1,0 ;MASK INHIBITED BITS
21 00452 024040 LDA 1,C17
22
23 00453 060300 FILL: NIOP 0 ;SYNC AT A74
24 00454 034060 LDA 3,BPROG ;CODE TO AVOID OVERWRITE
25 00455 156436 SUBZ# 2,3,SEZ ;OF THE PROGRAM.
26 00456 000403 JMP .+3 ;PATT < PROGRAM BEGIN.
27 00457 034061 LDA 3,EPROG
28 00460 172436 SUBZ# 3,2,SEZ ;DONT SKIP IF > PROG END.
29 00461 041000 STA 0,0,2
30 00462 034042 LDA 3,C77
31 00463 151400 INC 2,2 ;PATTERN
32 00464 133414 AND# 1,2,SZR ;SKIP EVERY 16 TIMES
33 00465 000766 JMP FILL
34 00466 157414 AND# 2,3,SZR ;SKIP EVERY 64 TIMES
35 00467 000760 JMP IPAT1
36 00470 020004 LDA 0,FINAL ;TEST FOR FINAL ADDRESS
37 00471 142432 SUBZ# 2,0,SZC ;EVERY 64 LOC. 4K
38 00472 000752 JMP IPAT+1 ;FILL TIME=100MS.
39
```

A 0011 .MAIN

```
01
02 00473 030051 DISTUR: LDA 2,MODUAL ;DISTURB MODULE SELECT
03 00474 020043 LDA 0,C7777 ;DISTURB AT LOCATION
04 00475 024052 LDA 1,EDIST ;0101,0202,0303,ETC.
05 00476 123000 ADD 1,0
06 00477 024045 LDA 1,C101 ;EVERY OTHER CORE IN MEMORY
07 00500 133000 ADD 1,2 ;IS DISTURBED AT LEAST
08 00501 074477 READS 3 ;1024 TIMES+INHIBIT DISTURBS.
09 00502 175112 MOVL# 3,3,SZC ;BUT ONLY IF SWITCH 0
10 00503 142433 SUBZ# 2,0,SNC ;IS SET TO A ONE.
11 00504 000406 JMP ICHECK ;END OF DISTURB
12 00505 176400 SUB 3,3
13 00506 025000 LDA 1,0,2 ;REFERENCE MEMORY
14 00507 175704 INCS 3,3,SZR
15 00510 000776 JMP .-2
16 00511 000766 JMP DISTURB+4
17
18 00512 030003 ICHECK: LDA 2,ADR ;INITIALIZE CHECK CYCLE
19 00513 024041 LDA 1,C400
20 00514 020006 LDA 0,PATT ;X LINE INIT PATTERN
21 00515 133414 AND# 1,2,SZR
22 00516 100000 ICK: COM 0,0
23 00517 034005 LDA 3,INH
24 00520 163400 AND 3,0 ;MASK INHIBITED BITS
25 00521 024047 LDA 1,CNIO5 ;"S" PULSE
26 00522 114044 COM0 0,3,SZR ;"C" PULSE
27 00523 024046 LDA 1,CNIOC ;ON 1/0 DISTURB SIGNALS
28 00524 044412 STA 1,CHECK
29
30
31 00525 024060 LDA 1,BPROG ;CODE TO PREVENT PATTERN
32 00526 146436 SUBZ# 2,1,SEZ ;CHECK OF PROGRAM.
33 00527 000407 JMP CHECK ;PATT<PROGRAM BEGIN.
34 00530 024061 LDA 1,EPROG
35 00531 132436 SUBZ# 1,2,SEZ
36 00532 000404 JMP CHECK ;PATT>PROGRAM END
37 00533 024064 LDA 1,C20
38 00534 133000 ADD 1,2
39 00535 000416 JMP ECHECK
40 00536 000000 CHECK: 0 ;A SYNC PULSE ISSUED
41 00537 025000 LDA 1,0,2 ;SIGNALS RWVZ,RWV1
42 00540 106414 SUB# 0,1,SZR
43 00541 004427 JSR ERR1
44 00542 055000 STA 3,0,2
45 00543 025000 LDA 1,0,2
46 00544 136414 SUB# 1,3,SZR ;SIGNALS UV1,UVZ
47 00545 004424 JSR ERR2
48 00546 041000 STA 0,0,2
49 00547 151400 INC 2,2
50 00550 024040 LDA 1,C17 ;COUNT 16 TIMES
51 00551 147414 AND# 2,1,SZR
52 00552 000764 JMP CHECK
53
54
```

A 0012 .MAIN

```

01
02 00553 034042 ECHECK: LDA 3,C77
03 00554 157414 AND# 2,3,SZR ICHECK FOR END OF
04 00555 000741 JMP ICK ILINE
05 00556 024004 LDA 1,FINAL IEVERY 64 TIMES
06 00557 146432 SUBZ# 2,1,SZC ICHECK FOR END OF CORE
07 00560 000733 JMP ICHECK+1
08 00561 020006 LDA 0,PATT ICOMP THE
09 00562 100000 COM 0,0 IPATTERN
10 00563 040006 STA 0,PATT
11 00564 101004 MOV 0,0,SZR
12 00565 000647 JMP BEGIN+2
13 00566 034066 LDA 3,RETURN
14 00567 001400 JMP 0,3

```

```

15
16
17
18 00570 101020 ERR1: MOVZ 0,0 IDISTURB ENTRY
19 00571 054007 ERR2: STA 3,ERET IUNDISTURB ENTRY
20 00572 034044 LDA 3,C207 IC(1)=ERROR WORD
21 00573 063411 SKPBN TTO IC(2)=ERROR ADDRESS
22 00574 075111 DOAS 3,TTO ISET SWITCH 1 TO
23 00575 074477 READS 3 IINHIBIT HALT
24 00576 175200 MOVR 3,3
25 00577 034057 LDA 3,PL0C
26 00600 054066 STA 3,RETURN
27 00601 101003 MOV 0,0,SNC
28 00602 063077 ER: HALT IIF TTY NOT BUSY
29 00603 114040 COMO 0,3 ITURN OF TTY IF
30 00604 002007 JMP 0ERET ITO NOISEY.
31 00605 000605 CEND: .

```

```

32
33 .TXT /COPYRIGHT (C) DGC,1970,71,74

```

```

00606 047503
00607 054520
00610 044522
00611 044107
00612 020124
00613 041450
00614 020051
00615 043504
00616 026103
00617 034461
00620 030067
00621 033454
00622 026061
00623 032067

```

34 00624 046101 ALL RIGHTS RESERVED/

```

00625 020114
00626 044522
00627 044107
00630 051524
00631 051040
00632 051505
00633 051105
00634 042526
00635 000104

```

```

35
36 .END

```

0013 .MAIN

ADR	000003	4/05	4/49	10/06	10/14	11/18			
BEGIN	000432	4/27	4/34	10/04	12/12				
BPROG	000060	4/29	4/56	10/24	11/31				
C0700	000050	4/21	10/07						
C101	000045	4/18	11/06						
C17	000040	4/13	10/21	11/50					
C20	000064	4/33	11/37						
C207	000044	4/17	12/20						
C212	000402	9/07	9/13						
C215	000403	9/05	9/14						
C377	000414	7/16	9/24						
C400	000041	4/14	10/15	11/19					
C60	000413	7/30	7/49	7/53	9/22				
C77	000042	4/15	10/30	12/02					
C7777	000043	4/16	11/03						
C8EG	000065	4/34	5/03						
CEND	000505	4/27	12/31						
CH240	000411	7/27	7/34	8/47	9/20				
CHAR	000343	7/23	7/43	7/56	8/25	9/06	9/08		
CHAR1	000356	8/37							
CHAR7	000410	8/44	9/19						
CHECK	000536	11/28	11/33	11/36	11/40	11/52			
CHORZ	000407	8/43	9/10	9/18	9/27				
CHRET	000405	8/25	8/41	8/46	9/16				
CHSAV	000406	8/28	8/34	9/17					
CHTAB	000404	7/41	8/37	9/15					
CMA	000055	4/26	6/29						
CNI0C	000046	4/19	11/27						
CNI0S	000047	4/20	11/25						
CRLF	000372	6/21	9/04						
CRLFR	000412	9/04	9/11	9/21					
DEC0C	000276	7/38	7/57						
DEC0T	000306	7/46	7/51						
DECP	000314	7/47	7/52						
DECTB	000332	7/33	8/12						
DISTU	000473	11/02	11/16						
ECHEC	000553	11/39	12/02						
EDIST	000052	4/23	10/10	11/04					
EPROG	000061	4/30	4/58	10/27	11/34				
ER	000602	12/28							
ERET	000007	4/09	12/19	12/30					
ERR1	000570	11/43	12/18						
ERR2	000571	11/47	12/19						
FILL	000453	10/23	10/33						
FINAL	000004	4/06	4/40	6/20	6/24	10/08	10/36	12/05	
ICHEC	000512	11/11	11/18	12/07					
ICK	000516	11/22	12/04						
INH	000005	4/07	10/19	11/23					
IPAT	000443	10/14	10/38						
IPAT1	000447	10/18	10/35						
K1000	000053	4/24	6/02						
M20	000063	4/32	4/47	4/54					
M201	000054	4/25	6/18						
MESIZ	000234	6/23	6/46						
MESS	000247	6/22	7/13	7/24					
MESSR	000415	7/13	7/14	7/25	9/25				
MMA	000212	5/12	6/27						
MMA1	000217	6/32	6/43						
MMA2	000224	4/26	6/37						



## 0015 .MAIN

MODUA	000051	4/22	10/12	11/02		
MOVE	000113	5/02	5/09			
MSIZ	000164	4/03	6/02			
MSIZ1	000166	6/04	6/16			
MSIZ2	000202	6/11	6/18			
OCTAB	000322	7/31	8/03			
PATT	000006	4/08	10/16	11/20	12/08	12/10
POEC	000271	7/33				
PLOC	000057	4/28	4/53	5/10	12/25	
POCT	000266	6/25	7/30			
PSIZE	000056	4/27	4/41			
RADRE	000342	7/35	7/40	8/23		
RAND0	000062	4/31	4/39			
RETUR	000066	4/35	10/05	12/13	12/26	
SEND	000067	4/38	6/44			
TYPE	000416	8/40	8/48	9/26		
TYPRE	000430	9/26	9/31	9/35	9/36	
ZOCT	000264	7/27				
ZSUPP	000341	7/36	7/44	7/54	8/22	
.RAND	000126	4/38	5/14			
.UD01	000156	5/16	5/24	5/40		
.UD02	000157	5/17	5/25	5/41		
.UD03	000160	5/14	5/15	5/26	5/42	
.UD10	000161	5/29	5/32	5/43		
.UD20	000162	5/21	5/44			
.UD21	000163	5/28	5/46			
.UD50	000143	5/20	5/28			

