DataGeneral

DIAGNOSTIC LISTING

LISTING

096-000467-00

PROGRAM

TURN KEY BINARY LOADER

TAPE

095-000467-00

```
 0001 .MAIN MACRO REV 04.00              10:34:18 12/03/76
01
02
03
04
05
06
07              /*****************************************************************
08              /
09              / NAME: TURNKYL.SR                          PART NUMBER: 094-000866
10              /
11              /
12              / DESCRIPTION: TURN KEY BINARY LOADER
13              /
14              /
15              / REVISION HISTORY:
16              /
17              /       REV.            DATE
18              /
19              /       00              12/03/76
20              /
21              /
22              / COPYRIGHT (C) DATA GENERAL CORPORATION, 1976
23              / ALL RIGHTS RESERVED.
24              /*****************************************************************
```

```
 0002 .MAIN
01
02              /
03              / PREAMBLE FOR NEW BOOT PROGRAM
04              /
05
06        000000            .LOC 0
07
08        061277            MNRST=DOAC 0 77
09        000027            GET=27
10        000014            DVCD=14
11
12 00000  000001            000001            /TAPE SYNCHRONIZER
13 00001  177744            BEG-END-2         /NEGATIVE WORD COUNT FOR PREAMBLE
14
15 00002  030431 BEG:       LDA 2,C4K-.,1     /FIST MEMORY SIZING LOCATION
16 00003  151400 LOOP:      INC 2,2
17 00004  021001            LDA 0,1,2         /SAVE
18 00005  100800            COM 0,1           /COMPLEMENT
19 00006  045001            STA 1,1,2         /WRITE
20 00007  035001            LDA 3,1,2         /READ
21 00010  041001            STA 0,1,2         /RESTORE
22 00011  151513            INCL# 2,2,SNC     /SKIP IF OVER 32K
23 00012  136404            SUB 1,3,SZR       /SKIP IF NOT END OF THE AVAILABLE
24                                            /MEMORY
25 00013  000402            JMP .+2-.,1
26 00014  000767            JMP LOOP-.,1      /CONTINUE SIZING
27 00015  004027            JSR GET           /GET
28 00016  044415            STA 1,C4K-.,1     /SAVE COUNT OF BINLOADER
29 00017  133000            ADD 1,2           /FORM FIRST ADDRESS
30 00020  151400            INC 2,2           /INCREMENT ADDRESS
31 00021  004027            JSR GET           /GET
32 00022  045000            STA 1,0,2         /SET INTO MEMORY
33 00023  010410            ISZ C4K-.,1       /BUMP COUNT
34 00024  000774            JMP .-4-.,1       /GO BACK
35 00025  020014            LDA 0,DVCD        /LOCATION OF PROG LOAD DEV CODE
36 00026  041371            STA 0,-7,2        /SAVE FOR BINARY LOADER
37 00027  062677            IORST
38 00030  061277            MNRST
39 00031  063077            HALT
40 00032  001000            JMP 0,2
41 00033  000376 C4K:       376
42 00034  000746 END:       JMP BEG-.,1               /GETS CONTROL HERE
```

```
10003 .MAIN
01
02
03                ;START
04                ;BINARY BLOCK LOADER
05
06                ;SUBROUTINE TO ASSEMBLE A WORD INTO AC2, THIS WORD IS
07                ;ADDED INTO THE CHECKSUM HELD IN AC0
08
09 00035 177623           BUILD=BEND=1      ;MINUS WORD COUNT FOR BIN LOADER
10
11 00036 054521 BUILD:    STA 3,TEMP1=.,1   ;SAVE THE RETURN
12 00037 004407           JSR GTCHR=.,1     ;GET CHARACTER INTO AC3
13 00040 171300           MOVS 3,2          ;AND SAVE IN THE LH OF AC2
14 00041 004405           JSR GTCHR=.,1     ;GET THE NEXT CHARACTER
15 00042 173300           ADDS 3,2          ;AND BUILD IN AC2
16 00043 143000           ADD 2,0           ;ADD INTO CHECKCUM
17 00044 002513           JMP @TEMP1=.,1    ;AND RETURN
18 00045 000004 DIFF:     4
19
20                ;SUBROUTNE TO GET A CHARACTER INTO AC3
21                ;IF SWITCH0=0, USE TELETYPE, ELSE USE PTR
22 00046 054512 GTCHR:    STA 3,TEMP2=.,1   ;SAVE THE RETURN
23 00047 000000 IO1:      0                 ;SKPDN DEV
24 00050 000777           JMP .-1=.,1
25 00051 000000 IO2:      0                 ;DIAS 3,DEV
26 00052 002506           JMP @TEMP2=.,1    ;RETURN
27
28                ;START OF THE LOADER
29 00053 062677 START:    IORST
30 00054 061277           MNRST
31 00055 020526           LDA 0,ADDRS-6-.,1       ;DEVICE CODE
32 00056 024506           LDA 1,C77=.,1     ;MASK FOR DEVICE CODE
33 00057 123400           AND 1,0           ;JUST DEVICE CODE
34 00060 024505           LDA 1,.IO1=.,1    ;ADDRESSS OF SKPDN
35 00061 107000           ADD 0,1           ;BUILD INSTRUCTION
36 00062 044765           STA 1,IO1=.,1     ;PUT THERE
37 00063 024503           LDA 1,.IO2=.,1
38 00064 107000           ADD 0,1
39 00065 044764           STA 1,IO2=.,1
40 00066 024501           LDA 1,.IO3=.,1
41 00067 107000           ADD 0,1
42 00070 044401           STA 1,IO3=.,1
43 00071 000000 IO3:      0                 ;START READER
44
45
```

```
10004 .MAIN
01
02
03                ;READ IN A BLOCK
04 00072 004754 BLOCK:    JSR GTCHR=.,1     ;GET A CHARACTER
05 00073 171305           MOVS 3,2,SNR      ;AND TEST IT FOR ZERO
06 00074 000776           JMP BLOCK=.,1     ;YES, STILL IN LEADER
07 00075 004751           JSR GTCHR=.,1     ;OK, BUILD A WORD
08 00076 173300           ADDS 3,2          ;IN AC2
09 00077 141000           MOV 2,0           ;SET INTO THE CHECKSUM
10 00100 145000           MOV 2,1           ;SET THE COUNTER
11 00101 004735           JSR BUILD=.,1     ;GO GET THE ADDRESS
12 00102 050507           STA 2,ADDRS=.,1   ;AND STORE IT
13 00103 004733           JSR BUILD=.,1     ;READ THE CHECKSUM WORD
14 00104 125113           MOVL# 1,1,SNC     ;TEST THE COUNT
15 00105 000431           JMP TEST=.,1      ;IT IS >0, IE A START OR IGNORE
16 00106 044454           STA 1,COUNT=.,1   ;BLOCK
17
18                ;READ IN THE DATA BLOCK
19 00107 030451           LDA 2,TEMP2=.,1   ;SEE IF STORAGE
20 00110 034735           LDA 3,DIFF=.,1
21 00111 172400           SUB 3,2
22 00112 034477           LDA 3,ADDRS=.,1   ;ADDRESS IS TOO BIG
23 00113 136400           SUB     1,3
24 00114 172023           ADCZ 3,2,SNC
25 00115 000415           JMP CHKER=.,1     ;YES, HALT THE LOADER
26 00116 030445           LDA     2,C20=.,1
27 00117 147033           ADDZ# 2,1,SNC
28 00120 010442           ISZ COUNT=.,1
29 00121 147022           ADDZ    2,1,SZC   ;REPEAT BLOCK?
30 00122 125113 STORE:    MOVL#   1,1,SNC
31 00123 004713           JSR BUILD=.,1
32 00124 052465           STA 2,@ADDRS=.,1
33 00125 010464           ISZ ADDRS=.,1
34 00126 010434           ISZ COUNT=.,1
35 00127 000773           JMP STORE=.,1
36 00130 105005           MOV 0,1,SNR       ;NOW, TEST THE CHECKSUM
37 00131 000741           JMP BLOCK=.,1     ;GO READ IN A BLOCK
38 00132 062677 CHKER:    IORST
39 00133 061277           MNRST
40 00134 063077           HALT              ;CHECKSUM ERROR, AC0=VALUE
41 00135 000735           JMP BLOCK=.,1
42
```

```
10005 .MAIN                                                                    0006 .MAIN
01                      ;START BLOCK OR IGNORE BLOCK
02 00136 125224 TEST:   MOVZR 1,1,SZR                                          **00000 TOTAL ERRORS, 00000 PASS 1 ERRORS
03 00137 000412         JMP IGNOR-.,1   ;AN IGNORE BLOCK
04 00140 105004         MOV 0,1,SZR     ;TEST THE CHECK SUM
05 00141 000771         JMP CHKER-.,1   ;ERROR
06 00142 030447         LDA 2,ADDRS-.,1 ;GET THE ADDRESS
07 00143 062677         IORST           ;DO A RESET
08 00144 061277         MNRST
09 00145 151113         MOVL# 2,2,SNC   ;TEST BIT 0
10 00146 001000         JMP 0,2         ;0=START THE PROGRAM
11 00147 063077         HALT            ;0, HALT
12 00150 000777         JMP .-1-.,1
13                      ;IGNORE ERROR MESSAGES BY READING UNTIL
14                      ;A RUBOUT
15
16 00151 004675 IGNOR:  JSR GTCHR-.,1   ;GET INTO AC3
17 00152 020404         LDA 0,C377-.,1
18 00153 116404         SUB 0,3,SZR
19 00154 000775         JMP IGNOR-.,1
20 00155 000715         JMP BLOCK-.,1   ;OK, GO INTO BLOCK MODE
21 00156 000377 C377:   377
22 00157 000000 TEMP1:  0
23 00160 000000 TEMP2:  0
24 00161 000000 SAVE:   0
25 00162 000000 COUNT:  0
26 00163 000020 C20:    20              ; REPEAT BLOCKS HAVE WD > 20(OCTAL)
27 00164 000077 C77:    77
28 00165 063600 .IO1:   SKPDN 0
29 00166 074500 .IO2:   DIAS 3,0
30 00167 060100 .IO3:   NIOS 0
31
32       000211         .LOC .+21       ;SKIP BOOTSTRAP (OLD NOVA)
33 00211 000000 ADDRS:  0
34 00212 000641 BEND:   JMP START-.,1
35
36
37 00213 047503         .TXT    /COPYRIGHT (C) DGC, 1976
38       054520
39       044522
40       044107
41       020124
42       041450
43       020051
44       043504
45       026103
46       030440
47       033471
48 00226 046466 ALL RIGHTS RESERVED./
49       046114
50       051040
51       043511
52       052110
53       020123
54       042522
55       042523
56       053122
57       042105
58       000056
59
60                      .END
```

0007 .MAIN

ADDRS	000211	3/31	4/12	4/22	4/32	4/33	5/06	5/33
BEG	000002	2/13	2/15	2/42				
BEND	000212	3/09	5/34					
BLOCK	000072	4/04	4/06	4/37	4/41	5/20		
BUILD	000036	3/09	3/11	4/11	4/13	4/31		
C20	000163	4/26	5/26					
C377	000156	5/17	5/21					
C4K	000033	2/15	2/28	2/33	2/41			
C77	000164	3/32	5/27					
CHKER	000132	4/25	4/38	5/05				
COUNT	000162	4/16	4/28	4/34	5/25			
DIFF	000045	3/18	4/20					
DVCO	000014	2/10	2/35					
END	000034	2/13	2/42					
GET	000027	2/09	2/27	2/31				
GTCHR	000046	3/12	3/14	3/22	4/04	4/07	5/16	
IGNOR	000151	5/03	5/16	5/19				
IO1	000047	3/23	3/36					
IO2	000051	3/25	3/39					
IO3	000071	3/42	3/43					
LOOP	000003	2/16	2/26					
MNRST	001277	2/08	2/38	3/30	4/39	5/08		
SAVE	000161	5/24						
START	000053	3/29	5/34					
STORE	000122	4/30	4/35					
TEMP1	000157	3/11	3/17	5/22				
TEMP2	000160	3/22	3/26	4/19	5/23			
TEST	000136	4/15	5/02					
.IO1	000165	3/34	5/28					
.IO2	000166	3/37	5/29					
.IO3	000167	3/40	5/30					